GATE OVERFLOW

Volume 2
Algorithms,
Programming & DS
Theory of
Computation
Compilers

42 GATE CSE papers
1987- 2021 covered
5 GATE IT papers
1080+ questions
1120+ answers

GATE Overflow for GATECSE

EDITION 3 FOR GATE 2022

GATE Overflow Team
led by

ARJUN SURESH

Computer Science & IT

QnA

Chat

FB Group

Table of Contents

Contributors

User	👍, Answers
Arjun Suresh	10924, 225
Praveen Saini	3127, 62
Gate Keeda	1884, 60
Akash Kanase	1249, 34
Vikrant Singh	980, 18
Rajesh Pradhan	922, 22
Amar Vashishth	900, 20
Digvijay	885, 27
Prashant Singh	842, 28
Debashish Deka	816, 10
gatecse	815, 21
Bhagirathi Nayak	773, 21
Rajarshi Sarkar	754, 27
Sankaranarayanan P.N	586, 19
Kalpna Bhargav	584, 11
Pooja Palod	546, 15
Rishabh Gupta	466, 3
Sachin Mittal	444, 8
Ahwan Mishra	427, 9
Pragy Agarwal	408, 6
Manu Thakur	395, 10
Ankit Rokde	384, 10
Anurag Semwal	359, 7
Monanshi Jain	333, 8
Mithlesh Upadhyay	324, 7
Abhilash Panicker	319, 5
minal	304, 7
Deepak Poonia	294, 11
jayendra	292, 9
Anoop Sonkar	269, 6
Akhil Nadh PC	267, 6
Sandeep_Uniyal	265, 7
ryan sequeira	242, 3
sriv_shubham	240, 5
Harsh181996	231, 2
Srinath Jayachandran	229, 2
Shaik Masthan	226, 8
suraj	222, 6

User	?Added
Kathleen Bankson	445
Jotheeswari	179
Arjun Suresh	147
Ishrat Jahan	124
makhdoom ghaya	77
gatecse	40
Rucha Shelke	30
Akash Kanase	26
Sandeep Singh	25
Madhav kumar	11
khush tak	8

User	Done
kenzou	207
Milicevic3306	172
Arjun Suresh	132
Lakshman Patel	117
Naveen Kumar	102
soujanyareddy13	88
gatecse	85
Shikha Mallick	35
Akash Dinkar	24
Pooja Khatri	21
Krithiga2101	20
S k Rawani	9
Ajay kumar soni	7
Shaik Masthan	5
adityaravishankar28	4
Manu Thakur	4
Subarna Das	4
Digvijay	3
Sukanya Das	3
Deepak Poonia	3
Jotheeswari	3

1 Algorithms (298)

Searching, Sorting, Hashing, Asymptotic worst case time and Space complexity, Algorithm design techniques: Greedy, Dynamic programming, and Divide-and-conquer, Graph search, Minimum spanning trees, Shortest paths.

Mark Distribution in Previous GATE

Year	2021-1	2021-2	2020	2019	2018	2017-1	2017-2	2016-1	2016-2	Minimum	Average	Maximum
1 Mark Count	3	2	3	2	0	2	2	3	3	0	2.2	3
2 Marks Count	3	4	4	2	4	2	3	2	3	2	3	4
Total Marks	9	10	11	6	8	6	8	7	9	**6**	**8.2**	**11**

1.1 Algorithm Design (7) top

1.1.1 Algorithm Design: GATE CSE 1992 | Question: 8 top https://gateoverflow.in/587

Let T be a Depth First Tree of a undirected graph G. An array P indexed by the vertices of G is given. $P[V]$ is the parent of vertex V, in T. Parent of the root is the root itself.

Give a method for finding and printing the cycle formed if the edge (u, v) of G not in T (i.e., $e \in G - T$) is now added to T.

Time taken by your method must be proportional to the length of the cycle.

Describe the algorithm in a PASCAL (C) – like language. Assume that the variables have been suitably declared.

gate1992 algorithms descriptive algorithm-design

Answer

1.1.2 Algorithm Design: GATE CSE 1994 | Question: 7 top https://gateoverflow.in/2503

GY

An array A contains n integers in locations $A[0], A[1], \dots A[n-1]$. It is required to shift the elements of the array cyclically to the left by K places, where $1 \leq K \leq n-1$. An incomplete algorithm for doing this in linear time, without using another array is given below. Complete the algorithm by filling in the blanks. Assume all variables are suitably declared.

```
min:=n;
i=0;
while _____ do
begin
    temp:=A[i];
    j:=i;
    while ____ do
    begin
        A[j]:=____;
        j:=(j+K) mod n;
       if j<min then
        min:=j;
    end;
    A[(n+i-K)mod n]:=____;
    i:=______;
end;
```

gate1994 algorithms normal algorithm-design fill-in-the-blanks

Answer

1.1.3 Algorithm Design: GATE CSE 2006 | Question: 17 top https://gateoverflow.in/978

An element in an array X is called a leader if it is greater than all elements to the right of it in X. The best algorithm to find all leaders in an array

A. solves it in linear time using a left to right pass of the array
B. solves it in linear time using a right to left pass of the array
C. solves it using divide and conquer in time $\Theta(n \log n)$
D. solves it in time $\Theta(n^2)$

gate2006-cse algorithms normal algorithm-design

Answer

1.1.4 Algorithm Design: GATE CSE 2006 | Question: 54 top https://gateoverflow.in/1832

Given two arrays of numbers $a_1, \dots, a_n$ and $b_1, \dots, b_n$ where each number is 0 or 1, the fastest algorithm to find the largest span (i, j) such that $a_i + a_{i+1} + \dots + a_j = b_i + b_{i+1} + \dots + b_j$ or report that there is not such span,

A. Takes $O(3^n)$ and $\Omega(2^n)$ time if hashing is permitted
B. Takes $O(n^3)$ and $\Omega(n^{2.5})$ time in the key comparison mode
C. Takes $\Theta(n)$ time and space
D. Takes $O(\sqrt{n})$ time only if the sum of the $2n$ elements is an even number

gate2006-cse algorithms normal algorithm-design time-complexity

Answer

1.1.5 Algorithm Design: GATE CSE 2014 Set 1 | Question: 37 top https://gateoverflow.in/1915

There are 5 bags labeled 1 to 5. All the coins in a given bag have the same weight. Some bags have coins of weight 10 gm, others have coins of weight 11 gm. I pick $1, 2, 4, 8, 16$ coins respectively from bags 1 to 5 Their total weight comes out to 323 gm. Then the product of the labels of the bags having 11 gm coins is ___.

gate2014-cse-set1 algorithms numerical-answers normal algorithm-design

Answer

1.1.6 Algorithm Design: GATE CSE 2019 | Question: 25 top https://gateoverflow.in/302823

Consider a sequence of 14 elements: $A = [-5, -10, 6, 3, -1, -2, 13, 4, -9, -1, 4, 12, -3, 0]$. The sequence sum $S(i, j) = \Sigma_{k=i}^{j} A[k]$. Determine the maximum of $S(i, j)$, where $0 \leq i \leq j < 14$. (Divide and conquer approach may be used.)

Answer: ___________

gate2019-cse numerical-answers algorithms algorithm-design

Answer

1.1.7 Algorithm Design: GATE CSE 2021 Set 1 | Question: 40 top https://gateoverflow.in/357411

Define R_n to be the maximum amount earned by cutting a rod of length n meters into one or more pieces of integer length and selling them. For $i > 0$, let $p[i]$ denote the selling price of a rod whose length is i meters. Consider the array of prices:

$$\mathrm{p}[1] = 1, \mathrm{p}[2] = 5, \mathrm{p}[3] = 8, \mathrm{p}[4] = 9, \mathrm{p}[5] = 10, \mathrm{p}[6] = 17, \mathrm{p}[7] = 18$$

Which of the following statements is/are correct about R_7?

A. $R_7 = 18$
B. $R_7 = 19$
C. R_7 is achieved by three different solutions
D. R_7 cannot be achieved by a solution consisting of three pieces

gate2021-cse-set1 multiple-selects algorithms algorithm-design

Answer

Answers: Algorithm Design

1.1.1 Algorithm Design: GATE CSE 1992 | Question: 8 top https://gateoverflow.in/587

✔ Union-Find Algorithm can be used to find the cycle.

Ref: http://www.geeksforgeeks.org/union-find/

References

👍 13 votes -- Rajarshi Sarkar (27.8k points)

1.1.2 Algorithm Design: GATE CSE 1994 | Question: 7 top

☛ https://gateoverflow.in/2503

✔ The inner loop is left rotating elements each separated by k elements. i.e. A[i+k] goes to A[i], A[i+2k] goes to A[i+k] and so on.

This loop stops, when A[i] gets assigned to some place which should be A[n-k+i]. (we do not need mod n here because i < k)

If n is a multiple of K, the inner loop iterates n/K times and outer loop iterates K times.

Otherwise inner loop iterates more than n/K times and correspondingly the outer loop gets adjusted using the min variable.

```
min:=n;
i=0;
while i < min do
begin
    temp:=A[i];
    j:=i;
    while (j != n+i-K)  do //we completed a cycle when this equals
    begin
        A[j]:= A[(j+K) mod n];
        j:=(j+K) mod n;
        if j<min then //updates the iteration count for i loop
            min:=j;
    end;
    A[(n+i-K)mod n]:=temp; //we do not need mod n, because i is <= K
    i:= i+1;
end;
```

C code for the problem is as follows:

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
int main()
{
    int *A;
    int n, i, j, K, min;
    time_t t;
    time(&t); //to get current time
    srand(t); //initialize random seed using current time
    printf("Please enter n and K: ");
    scanf("%d%d", &n, &K);
    A = malloc(n * sizeof(int));
    printf("Enter n elements: \n");
    for(i = 0; i < n; i++)
        A[i] = rand()%1000;
    printf("n elements are: \n");
    for(i = 0; i < n; i++)
        printf("%d ", A[i]);
    i = 0, min = n;
    while(i < min)
    {
        int temp = A[i];
        j = i;
        while(j != n+i-K)//we completed a cycle when this equals
        {
            A[j] = A[(j+K)%n];
            j = (j+K) % n;
            if(j < min) min = j;
        }
        A[n+i-K] = temp;
        i++;
    }
    printf("\nThe numbers left rotated by %d places are: \n", K);
    for(i = 0; i < n; i++)
        printf("%d ", A[i]);
    free(A);
}
```

👍 9 votes -- Arjun Suresh (330k points)

1.1.3 Algorithm Design: GATE CSE 2006 | Question: 17 top

https://gateoverflow.in/978

✔ **Option B.** We can move from right to left, while keeping a note of the maximum element so far (let's call it `current_max`).

- Starting from the rightmost element, we initialize our `current_max` with it, since the rightmost element will always be a leader.
- Moving from right to left, if an element x is greater than our `current_max`, then x is also a leader. Add this element to list of leaders (or simply print it). Set `current_max` to x and carry-on leftward.

Time Complexity would be $\Theta(n)$.

50 votes -- Madhur Rawat (2k points)

1.1.4 Algorithm Design: GATE CSE 2006 | Question: 54 top

https://gateoverflow.in/1832

✔ Answer is (C). Following algorithm would do.

Since array is binary, the max sum will go until n and so the sum difference of the two arrays can vary between $-n$ and n. We use array $start$ to keep the starting index of each possible sum (hence of size $2n+1$) and array end to keep the ending index (these two arrays work like hash tables and since we have only $2n+1$ possible keys, we can do a perfect hashing). So, our required solution will be $max(end[i] - start[i])$ provided both are assigned values.

The algorithm works as follows:

1. Initialize $diff$ array to contain the difference of sum of elements of array a and b. i.e., $diff[i] = \sum_{i=0}^{n} a[i] - b[i]$.
2. Now $diff[i]$ can have values from $-n$ to n which gives $2n+1$ possible values and the first occurrence of a $diff$ value marks the beginning of a span and the last occurrence marks the end. We use $start$ and end array for storing these two positions for the $2n+1$ possible values.
3. Now, the largest value of $end[i] - start[i]$ for any i, will be the largest span and the start of it will be $start[i]+1$, and end will be $end[i]$. If the span is starting from first position itself (arrays a and b have same first elements), then it will start from $start[i]$ itself.

```
#include <stdio.h>

#define size 100 //assume n is less than 100
int main()
{
    int n, a[size], b[size];
    int start[2*size+1], end[2*size+1];
    int sum1 = 0, sum2 = 0, i;
    int diff[size];
    printf("Enter n: ");
    scanf("%d", &n);
    for(i = 0; i < n; i++)
    {
        printf("Enter a[%d]: ", i);
        scanf("%d", &a[i]);
    }
    for(i = 0; i < n; i++)
    {
        printf("Enter b[%d]: ", i);
        scanf("%d", &b[i]);
    }

    for(i = 0; i < n; i++)
    {
        if(a[i]) sum1++;
        if(b[i]) sum2++;
        diff[i] = sum1 - sum2;
    }
    for(i = 0; i < 2*n; i++)
        start[i] = -1,end[i] = -1;
    start[n] = end[n] = 0;
    //initially sum is 0 at the beginning of array and
    //the first n-1 elements of start and end are used
    //if sum of A till ith element is less than sum of B till ith element
    for(i=0; i < n; i++)
    {
        if(start[diff[i] + n] == -1)//interested only in the first occurrence of diff[i]
            start[diff[i] + n] = i;
        end[diff[i] + n] = i;//interested in the last occurrence of diff[i]
    }
    int max = -1;
    int savei = -1; //savei is for storing the sum having the largest span

    for(i = 0; i < 2*n; i++)
    {
        if(start[i] > -1 && (end[i] - start[i] > max))
        {
            max = end[i] - start[i];
            savei = i;
```

```
        }

    }
    if(savei >= 0)
    {
        printf("The largest span is from %d to %d\n", start[savei]+(savei != n), end[savei]);
    //when sum zero is having the largest span, span starts from first element itself.
    //Else, the span starts from the next element from which the span does not change
    }
    else
    {
        printf("No span\n");
    }
}
```

38 votes -- Arjun Suresh (330k points)

1.1.5 Algorithm Design: GATE CSE 2014 Set 1 | Question: 37 top https://gateoverflow.in/1915

Suppose X is the number of coins of 11 gm and Y is the number of 10 gm coins.
According to question,

$$11X + 10Y = 323 \quad \rightarrow (1)$$
$$X + Y = 31 \quad \rightarrow (2)$$

Solving $(1), (2)$, we get $X = 13$ and $Y = 18$. So, here number of coins of 11 gm is 13 and the only possible combination for 13 coins is

- 1 coin from bag 1
- 4 coins from bag 3
- 8 coins from bag 4

So, product of label of bags will be $= 1 \times 3 \times 4 = 12.$

99 votes -- minal (13.1k points)

1.1.6 Algorithm Design: GATE CSE 2019 | Question: 25 top https://gateoverflow.in/302823

Subsequence : A subsequence is a sequence that can be derived from another sequence by deleting some or no elements without changing the order of the remaining elements.

Subarray : A subarray of n-element array is an array composed from a contiguous block of the original array's elements.

Question is asking for subsequence.

$S(i,j) = \sum_{k=i}^{j} A[k]$

What does $\sum_{k=i}^{j}$ mean? It means you start from index i and go till index j with unit increment each time.

Ultimately they are asking 'Maximum subarray sum'.

```
int maxsum = A[0], sum = A[0], n=14;
for(int i=1; i<n; i++){
    if (A[i]>A[i-1]) {
        if (sum > 0) sum+=A[i];
        else sum = A[i];
    }
    else sum = A[i];
    if (maxsum < sum) maxsum = sum;
}
```

$\text{sum} = 6 + 3 - 1 - 2 + 13 + 4 - 9 - 1 + 4 + 12 = 29$

23 votes -- Digvijay (44.9k points)

1.1.7 Algorithm Design: GATE CSE 2021 Set 1 | Question: 40 top https://gateoverflow.in/357411

A & C should be correct

- 1^{st} Solution : $p[2]; p[3]; p[2] = 5 + 8 + 5 = 18$
- 2^{nd} Solution : $p[7] = 18$
- 3^{rd} Solution : $p[6]; p[1] = 17 + 1 = 18$

10 votes -- Tushar Agarwal (239 points)

1.2 Algorithm Design Techniques (7) top

1.2.1 Algorithm Design Techniques: GATE CSE 1990 | Question: 12b top https://gateoverflow.in/86146

Consider the following problem. Given n positive integers $a_1, a_2 \dots a_n$, it is required to partition them in to two parts A and B such that, $\left| \sum_{i \in A} a_i - \sum_{i \in B} a_i \right|$ is minimised

Consider a greedy algorithm for solving this problem. The numbers are ordered so that $a_1 \geq a_2 \geq \dots a_n$, and at i^{th} step, a_i is placed in that part whose sum in smaller at that step. Give an example with $n = 5$ for which the solution produced by the greedy algorithm is not optimal.

gate1990 descriptive algorithms algorithm-design-techniques

Answer

1.2.2 Algorithm Design Techniques: GATE CSE 1990 | Question: 2-vii top https://gateoverflow.in/83991

Match the pairs in the following questions:

(a)	Strassen's matrix multiplication algorithm	(p)	Greedy method
(b)	Kruskal's minimum spanning tree algorithm	(q)	Dynamic programming
(c)	Biconnected components algorithm	(r)	Divide and Conquer
(d)	Floyd's shortest path algorithm	(s)	Depth-first search

a -r
b - p
c - s
d - q

gate1990 match-the-following algorithms algorithm-design-techniques

Answer

1.2.3 Algorithm Design Techniques: GATE CSE 1995 | Question: 1.5 top https://gateoverflow.in/2592

Merge sort uses:

A. Divide and conquer strategy
B. Backtracking approach
C. Heuristic search
D. Greedy approach

gate1995 algorithms sorting easy algorithm-design-techniques merge-sort

Answer

1.2.4 Algorithm Design Techniques: GATE CSE 1997 | Question: 1.5 top https://gateoverflow.in/2221

The correct matching for the following pairs is

A.	All pairs shortest path	1.	Greedy
B.	Quick Sort	2.	Depth-First Search
C.	Minimum weight spanning tree	3.	Dynamic Programming
D.	Connected Components	4.	Divide and Conquer

A. A-2 B-4 C-1 D-3
B. A-3 B-4 C-1 D-2
C. A-3 B-4 C-2 D-1
D. A-4 B-1 C-2 D-3

gate1997 algorithms normal algorithm-design-techniques

Answer

1.2.5 Algorithm Design Techniques: GATE CSE 2015 Set 1 | Question: 6 top https://gateoverflow.in/8088

Match the following:

P.	Prim's algorithm for minimum spanning tree	i.	Backtracking
Q.	Floyd-Warshall algorithm for all pairs shortest path	ii.	Greedy method
R.	Merge sort	iii.	Dynamic programming
S.	Hamiltonian circuit	iv.	Divide and conquer

A. P-iii, Q-ii, R-iv, S-i
B. P-i, Q-ii, R-iv, S-iii
C. P-ii, Q-iii, R-iv, S-i
D. P-ii, Q-i, R-iii, S-iv

gate2015-cse-set1 algorithms normal algorithm-design-techniques

Answer

1.2.6 Algorithm Design Techniques: GATE CSE 2015 Set 2 | Question: 36 top https://gateoverflow.in/8161

Given below are some algorithms, and some algorithm design paradigms.

1.	Dijkstra's Shortest Path	i.	Divide and Conquer
2.	Floyd-Warshall algorithm to compute all pairs shortest path	ii.	Dynamic Programming
3.	Binary search on a sorted array	iii.	Greedy design
4.	Backtracking search on a graph	iv.	Depth-first search
		v.	Breadth-first search

Match the above algorithms on the left to the corresponding design paradigm they follow.

A. 1-i, 2-iii, 3-i, 4-v
B. 1-iii, 2-iii, 3-i, 4-v
C. 1-iii, 2-ii, 3-i, 4-iv
D. 1-iii, 2-ii, 3-i, 4-v

gate2015-cse-set2 algorithms easy algorithm-design-techniques

Answer

1.2.7 Algorithm Design Techniques: GATE CSE 2017 Set 1 | Question: 05 top https://gateoverflow.in/118707

Consider the following table:

Algorithms		Design Paradigms	
P.	Kruskal	i.	Divide and Conquer
Q.	Quicksort	ii.	Greedy
R.	Floyd-Warshall	iii.	Dynamic Programming

Match the algorithms to the design paradigms they are based on.

A. $(P) \leftrightarrow (ii), (Q) \leftrightarrow (iii), (R) \leftrightarrow (i)$
B. $(P) \leftrightarrow (iii), (Q) \leftrightarrow (i), (R) \leftrightarrow (ii)$
C. $(P) \leftrightarrow (ii), (Q) \leftrightarrow (i), (R) \leftrightarrow (iii)$
D. $(P) \leftrightarrow (i), (Q) \leftrightarrow (ii), (R) \leftrightarrow (iii)$

gate2017-cse-set1 algorithms algorithm-design-techniques

Answer

Answers: Algorithm Design Techniques

1.2.1 Algorithm Design Techniques: GATE CSE 1990 | Question: 12b top ☛ https://gateoverflow.in/86146

Let $S = \{12, 11, 8, 8, 8\}$

As per the given greedy algorithm we get $A = \{12, 8\}$ and $B = \{11, 8, 8\}$ and $|\sum_{i \in A} a_i - \sum_{i \in B} a_i| = |20 - 27| = 7.$

This is not minimal because if we partition S as $A = \{12, 11\}$ and $B = \{8, 8, 8\}$ we get $|\sum_{i \in A} a_i - \sum_{i \in B} a_i| = |23 - 24| = 1.$

Thus greedy algorithm fails for this example.

Working algorithm: https://www.geeksforgeeks.org/partition-a-set-into-two-subsets-such-that-the-difference-of-subset-sums-is-minimum/

References

12 votes -- Arjun Suresh (330k points)

1.2.2 Algorithm Design Techniques: GATE CSE 1990 | Question: 2-vii top ☛ https://gateoverflow.in/83991

✔ (a) Strassen's matrix multiplication algorithm - (r) Divide and Conquer

(b) Kruskal's minimum spanning tree algorithm - (p) Greedy method

(c) Biconnected components algorithm - (s) Depth first search

(d) Floyd's shortest path algorithm - (q) Dynamic programming

(a)	Strassen's matrix multiplication algorithm	(r)	Divide and Conquer
(b)	Kruskal's minimum spanning tree algorithm	(p)	Greedy method
(c)	Biconnected components algorithm	(s)	Depth-first search
(d)	Floyd's shortest path algorithm	(q)	Dynamic programming

18 votes -- vishwa ratna (1.7k points)

1.2.3 Algorithm Design Techniques: GATE CSE 1995 | Question: 1.5 top ☛ https://gateoverflow.in/2592

✔ **Answer: Option A.**

One of the best examples of **Divide and Conquer** strategy.

25 votes -- Gate Keeda (15.9k points)

1.2.4 Algorithm Design Techniques: GATE CSE 1997 | Question: 1.5 top ☛ https://gateoverflow.in/2221

✔ Answer : (B) A-3 B-4 C-1 D-2

A.	All pairs shortest path	3.	Dynamic Programming
B.	Quick Sort	2.	Divide and Conquer
C.	Minimum weight spanning tree	1.	Greedy
D.	Connected Components	2.	Depth-First Search

Reference: Read the Intro/Algo Sub-Heading.

- https://en.wikipedia.org/wiki/Floyd%E2%80%93Warshall_algorithm#History_and_naming
- https://en.wikipedia.org/wiki/Quicksort#Algorithm
- https://en.wikipedia.org/wiki/Minimum_spanning_tree#Algorithms
- https://en.wikipedia.org/wiki/Connected_component_(graph_theory)#Algorithms

References

👍 21 votes -- Siddharth Mahapatra (1.2k points)

1.2.5 Algorithm Design Techniques: GATE CSE 2015 Set 1 | Question: 6 top

☛ https://gateoverflow.in/8088

✔ P. Prim - ii. Greedy
http://www.geeksforgeeks.org/greedy-algorithms-set-5-prims-minimum-spanning-tree-mst-2/
Q. Floyd Warshall - iii. Dynamic
http://www.geeksforgeeks.org/dynamic-programming-set-16-floyd-warshall-algorithm/
R. Merge Sort - iv. Divide & Conquer
http://www.geeksforgeeks.org/merge-sort/
S. Hamiltonian Circuit - i. Backtracking
http://www.geeksforgeeks.org/backtracking-set-7-hamiltonian-cycle/

Option is C.

References

👍 15 votes -- Pronomita Dey (1.7k points)

1.2.6 Algorithm Design Techniques: GATE CSE 2015 Set 2 | Question: 36 top

☛ https://gateoverflow.in/8161

✔ Option C.

Dijkstra's Shortest path algorithm uses greedy design (always chosing the shortest neighbour) while Floyd Warshall Algorithm to compute all shortest paths uses Dynamic Programming.
Binary search uses Divide and Conquer (though we eliminate one part at each time) and Back tracking traversal to a graph uses Depth First Search (DFS) (in DFS we have to backtrack to a node after searching through all its descendant nodes if the searched node is not found).

👍 19 votes -- Raghuveer Dhakad (1.6k points)

1.2.7 Algorithm Design Techniques: GATE CSE 2017 Set 1 | Question: 05 top

☛ https://gateoverflow.in/118707

✔ In **Kruskal**, in every iteration, an edge of the **most minimum weight (greediest)** possible is selected and added to MST construction. Hence, **greedy**.

In **Quick Sort,** we partition the problem into subproblems, solve them and then combine. Hence, it is **Divide & Conquer**.

Floyd-Warshall uses Dynamic programming.

Hence, correct answer is : **OPTION (C).**

👍 25 votes -- sriv_shubham (2.8k points)

1.3 Asymptotic Notations (17) top

1.3.1 Asymptotic Notations: GATE CSE 1994 | Question: 1.23 top

☛ https://gateoverflow.in/2466

Consider the following two functions:

$$g_1(n) = \begin{cases} n^3 \text{ for } 0 \leq n \leq 10,000 \\ n^2 \text{ for } n \geq 10,000 \end{cases}$$

$$g_2(n) = \begin{cases} n \text{ for } 0 \leq n \leq 100 \\ n^3 \text{ for } n > 100 \end{cases}$$

Which of the following is true?

A. $g_1(n)$ is $O(g_2(n))$

B. $g_1(n)$ is $O(n^3)$
C. $g_2(n)$ is $O(g_1(n))$
D. $g_2(n)$ is $O(n)$

gate1994 algorithms asymptotic-notations normal multiple-selects

Answer

1.3.2 Asymptotic Notations: GATE CSE 1996 | Question: 1.11 top https://gateoverflow.in/2715

Which of the following is false?

A. $100n \log n = O(\frac{n \log n}{100})$
B. $\sqrt{\log n} = O(\log \log n)$
C. If $0 < x < y$ then $n^x = O(n^y)$
D. $2^n \neq O(nk)$

gate1996 algorithms asymptotic-notations normal

Answer

1.3.3 Asymptotic Notations: GATE CSE 1999 | Question: 2.21 top https://gateoverflow.in/1498

If $T_1 = O(1)$, give the correct matching for the following pairs:

(M) $T_n = T_{n-1} + n$	(U) $T_n = O(n)$
(N) $T_n = T_{n/2} + n$	(V) $T_n = O(n \log n)$
(O) $T_n = T_{n/2} + n \log n$	(W) $T_n = O(n^2)$
(P) $T_n = T_{n-1} + \log n$	(X) $T_n = O(\log^2 n)$

A. M-W, N-V, O-U, P-X
B. M-W, N-U, O-X, P-V
C. M-V, N-W, O-X, P-U
D. M-W, N-U, O-V, P-X

gate1999 algorithms recurrence asymptotic-notations normal

Answer

1.3.4 Asymptotic Notations: GATE CSE 2000 | Question: 2.17 top https://gateoverflow.in/664

Consider the following functions

- $f(n) = 3n^{\sqrt{n}}$
- $g(n) = 2^{\sqrt{n}\log_2 n}$
- $h(n) = n!$

Which of the following is true?

A. $h(n)$ is $O(f(n))$
B. $h(n)$ is $O(g(n))$
C. $g(n)$ is not $O(f(n))$
D. $f(n)$ is $O(g(n))$

gate2000-cse algorithms asymptotic-notations normal

Answer

1.3.5 Asymptotic Notations: GATE CSE 2001 | Question: 1.16 top https://gateoverflow.in/709

Let $f(n) = n^2 \log n$ and $g(n) = n(\log n)^{10}$ be two positive functions of n. Which of the following statements is correct?

A. $f(n) = O(g(n))$ and $g(n) \neq O(f(n))$
B. $g(n) = O(f(n))$ and $f(n) \neq O(g(n))$
C. $f(n) \neq O(g(n))$ and $g(n) \neq O(f(n))$
D. $f(n) = O(g(n))$ and $g(n) = O(f(n))$

gate2001-cse algorithms asymptotic-notations time-complexity normal

Answer

1.3.6 Asymptotic Notations: GATE CSE 2003 | Question: 20 top

https://gateoverflow.in/910

Consider the following three claims:

I. $(n+k)^m = \Theta(n^m)$ where k and m are constants
II. $2^{n+1} = O(2^n)$
III. $2^{2n+1} = O(2^n)$

Which of the following claims are correct?

A. I and II
B. I and III
C. II and III
D. I, II, and III

gate2003-cse algorithms asymptotic-notations normal

Answer

1.3.7 Asymptotic Notations: GATE CSE 2004 | Question: 29 top

https://gateoverflow.in/1026

The tightest lower bound on the number of comparisons, in the worst case, for comparison-based sorting is of the order of

A. n
B. n^2
C. $n \log n$
D. $n \log^2 n$

gate2004-cse algorithms sorting asymptotic-notations easy

Answer

1.3.8 Asymptotic Notations: GATE CSE 2005 | Question: 37 top

https://gateoverflow.in/1373

Suppose $T(n) = 2T(\frac{n}{2}) + n$, $T(0) = T(1) = 1$

Which one of the following is FALSE?

A. $T(n) = O(n^2)$
B. $T(n) = \Theta(n \log n)$
C. $T(n) = \Omega(n^2)$
D. $T(n) = O(n \log n)$

gate2005-cse algorithms asymptotic-notations recurrence normal

Answer

1.3.9 Asymptotic Notations: GATE CSE 2008 | Question: 39 top

https://gateoverflow.in/450

Consider the following functions:

- $f(n) = 2^n$
- $g(n) = n!$
- $h(n) = n^{\log n}$

Which of the following statements about the asymptotic behavior of $f(n)$, $g(n)$ and $h(n)$ is true?

A. $f(n) = O(g(n)); g(n) = O(h(n))$
B. $f(n) = \Omega(g(n)); g(n) = O(h(n))$
C. $g(n) = O(f(n)); h(n) = O(f(n))$
D. $h(n) = O(f(n)); g(n) = \Omega(f(n))$

gate2008-cse algorithms asymptotic-notations normal

Answer

1.3.10 Asymptotic Notations: GATE CSE 2011 | Question: 37 top https://gateoverflow.in/2139

Which of the given options provides the increasing order of asymptotic complexity of functions f_1, f_2, f_3 and f_4?

- $f_1(n) = 2^n$
- $f_2(n) = n^{3/2}$
- $f_3(n) = n \log_2 n$
- $f_4(n) = n^{\log_2 n}$

A. f_3, f_2, f_4, f_1
B. f_3, f_2, f_1, f_4
C. f_2, f_3, f_1, f_4
D. f_2, f_3, f_4, f_1

gate2011-cse algorithms asymptotic-notations normal

Answer

1.3.11 Asymptotic Notations: GATE CSE 2012 | Question: 18 top https://gateoverflow.in/50

Let $W(n)$ and $A(n)$ denote respectively, the worst case and average case running time of an algorithm executed on an input of size n. Which of the following is **ALWAYS TRUE**?

A. $A(n) = \Omega(W(n))$
B. $A(n) = \Theta(W(n))$
C. $A(n) = \mathrm{O}(W(n))$
D. $A(n) = \mathrm{o}(W(n))$

gate2012-cse algorithms easy asymptotic-notations

Answer

1.3.12 Asymptotic Notations: GATE CSE 2015 Set 3 | Question: 4 top https://gateoverflow.in/8398

Consider the equality $\sum_{i=0}^{n} i^3 = X$ and the following choices for X:

I. $\Theta(n^4)$
II. $\Theta(n^5)$
III. $O(n^5)$
IV. $\Omega(n^3)$

The equality above remains correct if X is replaced by

A. Only I
B. Only II
C. I or III or IV but not II
D. II or III or IV but not I

gate2015-cse-set3 algorithms asymptotic-notations normal

Answer

1.3.13 Asymptotic Notations: GATE CSE 2015 Set 3 | Question: 42 top https://gateoverflow.in/8501

Let $f(n) = n$ and $g(n) = n^{(1+\sin\ n)}$, where n is a positive integer. Which of the following statements is/are correct?

I. $f(n) = O(g(n))$
II. $f(n) = \Omega(g(n))$

A. Only I
B. Only II
C. Both I and II
D. Neither I nor II

gate2015-cse-set3 algorithms asymptotic-notations normal

Answer

1.3.14 Asymptotic Notations: GATE CSE 2017 Set 1 | Question: 04 top https://gateoverflow.in/118703

Consider the following functions from positive integers to real numbers:

$10, \sqrt{n}, n, \log_2 n, \frac{100}{n}$.
The CORRECT arrangement of the above functions in increasing order of asymptotic complexity is:

A. $\log_2 n, \frac{100}{n}, 10, \sqrt{n}, n$
B. $\frac{100}{n}, 10, \log_2 n, \sqrt{n}, n$
C. $10, \frac{100}{n}, \sqrt{n}, \log_2 n, n$
D. $\frac{100}{n}, \log_2 n, 10, \sqrt{n}, n$

gate2017-cse-set1 algorithms asymptotic-notations normal

Answer

1.3.15 Asymptotic Notations: GATE CSE 2021 Set 1 | Question: 3 top https://gateoverflow.in/357449

Consider the following three functions.

$$f_1 = 10^n \quad f_2 = n^{\log n} \quad f_3 = n^{\sqrt{n}}$$

Which one of the following options arranges the functions in the increasing order of asymptotic growth rate?

A. f_3, f_2, f_1
B. f_2, f_1, f_3
C. f_1, f_2, f_3
D. f_2, f_3, f_1

gate2021-cse-set1 algorithms asymptotic-notations

Answer

1.3.16 Asymptotic Notations: GATE IT 2004 | Question: 55 top https://gateoverflow.in/3698

Let $f(n)$, $g(n)$ and $h(n)$ be functions defined for positive integers such that
$f(n) = O(g(n)), g(n) \neq O(f(n)), g(n) = O(h(n))$, and $h(n) = O(g(n))$.
Which one of the following statements is FALSE?

A. $f(n) + g(n) = O(h(n) + h(n))$
B. $f(n) = O(h(n))$
C. $h(n) \neq O(f(n))$
D. $f(n)h(n) \neq O(g(n)h(n))$

gate2004-it algorithms asymptotic-notations normal

Answer

1.3.17 Asymptotic Notations: GATE IT 2008 | Question: 10 top https://gateoverflow.in/3270

Arrange the following functions in increasing asymptotic order:

a. $n^{1/3}$
b. e^n

c. $n^{7/4}$
d. $n \log^9 n$
e. 1.0000001^n

A. a, d, c, e, b
B. d, a, c, e, b
C. a, c, d, e, b
D. a, c, d, b, e

gate2008-it algorithms asymptotic-notations normal

Answer

Answers: Asymptotic Notations

1.3.1 Asymptotic Notations: GATE CSE 1994 | Question: 1.23 top

https://gateoverflow.in/2466

✔ For asymptotic complexity, we assume sufficiently large n. So, $g_1(n) = n^2$ and $g_2(n) = n^3$. Growth rate of g_1 is less than that of g_2, i.e., $g_1(n) = O(g_2(n))$.

Options A and B are TRUE here.

53 votes -- Arjun Suresh (330k points)

1.3.2 Asymptotic Notations: GATE CSE 1996 | Question: 1.11 top

https://gateoverflow.in/2715

✔

A. $100n \log n = O(\frac{n \log n}{100})$ **:** Big-O denotes the growth rate of functions and multiplication or division by a constant does not change the growth rate. So, this is TRUE and here O can even be replaced by Θ or Ω.
B. $\sqrt{\log n} = O(\log \log n)$ **:** FALSE. $\sqrt{\log n} = (\log n)^{0.5}$ grows faster than $\log \log n$ as any positive polynomial function (including powers between $0 - 1$) grows faster than any polylogarithmic function. (Ref: Section 3.2- Logarithms, Cormen). We can also do substitution here, but we must do for at least 2 large values and ensure it works for any larger n also.
C. $0 < x < y$ then $n^x = O(n^y)$ **:** TRUE since y is always greater than x. So, RHS is always greater than LHS.
D. $2^n \neq O(nk)$ **:** TRUE since k is constant. So, for large values of n, LHS is much higher than RHS (exponential function always greater than linear).

Only **B** is FALSE.

47 votes -- Prashant Singh (47.1k points)

1.3.3 Asymptotic Notations: GATE CSE 1999 | Question: 2.21 top

https://gateoverflow.in/1498

✔ (M) $T(n)$ = Sum of first n natural numbers = $\frac{n(n+1)}{2} = O(n^2)$

(N) $T(n) = \Theta(n) = O(n)$, third case of Master theorem

$(f(n) = n = \Omega\left(n^{\log_b a+\epsilon}\right) = \Omega\left(n^{\log_2 1+\epsilon}\right) = \Omega\left(n^{0+\epsilon}\right)$, satisfied for any positive $\epsilon \leq 1$. Also, $af\left(\frac{n}{b}\right) < cf(n) \implies f\left(\frac{n}{2}\right) < cf(n) \implies \frac{n}{2} < cn$, satisfied for any c greater than 0.5)

(O) $T(n) = \Theta(n \log n) = O(n \log n)$, third case of Master theorem

$(f(n) = n \log n = \Omega\left(n^{\log_b a+\epsilon}\right) = \Omega\left(n^{\log_2 1+\epsilon}\right) = \Omega\left(n^{0.5+\epsilon}\right)$, satisfied for positive $\epsilon = 0.5$. Also, $af\left(\frac{n}{b}\right) < cf(n) \implies f\left(\frac{n}{2}\right) < cf(n) \implies \frac{n}{2} \log \frac{n}{2} < cn \log n$, satisfied for $c = 0.5$)

(P) Like in (M), here we are adding the log of the first n natural numbers. So,

$T_n = \log 1 + \log 2 + \log 3 + \dots + \log n$

$= \log(1 \times 2 \times \dots \times n)$

$= \log(n!)$

$= \Theta(n \log n)$ (Stirling's Approximation)

👍 46 votes -- Arjun Suresh (330k points)

1.3.4 Asymptotic Notations: GATE CSE 2000 | Question: 2.17 top

☛ https://gateoverflow.in/664

✔

	$\mathbf{n= 256}$	$\mathbf{n = 65536}$
$f(n) = 3n^{\sqrt{n}}$	3×256^{16} $= 3 \times 2^{128}$	3×65536^{256} $= 3 \times 2^{16 \times 256}$ $= 3 \times 2^{4096}$
$g(n) = 2^{\sqrt{n} \log_2 n}$	$2^{16 \times 8}$ $= 2^{128}$	$2^{256 \times 16}$ $= 2^{4096}$
$h(n) = n!$	$256!$ $= O\left(\left(2^8\right)^{256}\right)$ $= O\left(2^{2048}\right)$	$65536!$ $= O\left(\left(2^{16}\right)^{65536}\right)$ $= O\left(2^{1M}\right)$

Case of $h(n)$ is given only by an upper bound but factorial has higher growth rate than exponential.

http://math.stackexchange.com/questions/351815/do-factorials-really-grow-faster-than-exponential-functions

$f(n)$ and $g(n)$ are having same order of growth as $f(n)$ is simply $3 \times g(n)$ (we can prove this by taking $\log$ also). So, (**d**) is correct and all other choices are false.

References

👍 60 votes -- Arjun Suresh (330k points)

1.3.5 Asymptotic Notations: GATE CSE 2001 | Question: 1.16 top

☛ https://gateoverflow.in/709

✔ A more formal approach:

$f(n) = n^2 \log n$

$g(n) = n(\log n)^{10}$

We can use the limit definition of O-notation

http://cse.unl.edu/~choueiry/S06-235/files/Asymptotics-HandoutNoNotes.pdf

$$\lim_{n\to\infty} \frac{f(n)}{g(n)} = 0, \implies f(n) = o(g(n))$$

small o implying f is strictly asymptotically lower than g. Also by definition, $o \implies O$ but $O \not\Longrightarrow o$.

$$\lim_{n\to\infty} \frac{f(n)}{g(n)} = c, c > 0 \implies f(n) = \Theta(g(n))$$

$$\lim_{n\to\infty} \frac{f(n)}{g(n)} = \infty, \implies f(n) = \omega(g(n))$$

small ω implying f is strictly asymptotically higher than g. Also by definition, $\omega \implies \Omega$, but $\Omega \not\Longrightarrow \omega$.

We can use this to prove the above question

For any $k > 0$

$$\lim_{n\to\infty} \frac{(\log n)^k}{n}$$

Applying L'Hôpital's rule,

$$\implies \lim_{n\to\infty} \frac{k * (\log n)^{k-1}}{n}$$

$$= \lim_{n\to\infty} \frac{k!}{n} = 0$$

So, $(\log n)^k = o(n)$

Now for large n, $n >> (\log n)^9$

i.e., $n^2 \log n >> n(\log n)^{10}$ (Multiplying both sides by $n \log n$)

So, $n(\log n)^{10} = o(n^2 \log n)$ and $n(\log n)^{10} \neq \Theta(n^2 \log n)$ (making LHS strictly asymptotically lower than RHS)

Or

$n(\log n)^{10} = O(n^2 \log n)$ and $n^2 \log n \neq O(n(\log n)^{10})$

Option B.

References

33 votes -- janakyMurthy (733 points)

	$f(n)$	$g(n)$
$n = 2^{10}$	$10 \times 2^{10} \times 2^{10}$	$2^{10} \times 10^{10}$
$n = 2^{256}$	$256 \times 2^{256} \times 2^{256}$	$2^{256} \times 256^{10}$

As n is going larger, $f(n)$ is overtaking $g(n)$ and the growth rate of f is faster than that of g. So, $g(n) = O(f(n))$ and $f(n) \neq O(g(n))$.

B choice.

81 votes -- Arjun Suresh (330k points)

1.3.6 Asymptotic Notations: GATE CSE 2003 | Question: 20 top https://gateoverflow.in/910

✔ I. Rate of growth of $(n+k)^m$ is same as that of (n^m) as k and m are constants. (If either k or m is a variable then the equality does not hold), i.e., for sufficiently large values of n,

$$(n+k)^m \leq an^m \text{ and}$$

$$n^m \leq b(n+k)^m$$

where a and b are positive constants. Here, a can be $k^m$and b can be 1.

So, **TRUE**.

II. $2^{n+1} = 2 \times (2^n) = \Theta\left(2^n\right)$ as 2 is a constant here.

As 2^{n+1} is both upper and lower bounded by 2^n we can say $2^{n+1} = O\left(2^n\right)$. ($\Theta$ implies both O as well as Ω)

So, **TRUE**.

III. 2^{2n+1} has same rate of growth as 2^{2n}.

$2^{2n} = 2^{n^2} = 2^n \times 2^n$

2^n is upper bounded by $(2^n)^2$, not the other way round as 2^{2n} is increasing by a factor of 2^n which is not a constant.

So, **FALSE**.

Correct Answer: A

45 votes -- Danish (3.4k points)

1.3.7 Asymptotic Notations: GATE CSE 2004 | Question: 29 top https://gateoverflow.in/1026

✔ For comparison-based sorting the asymptotically tight bound for worst case is given by $\Theta(n \log n)$, which means it is the tightest upper bound (big O) as well as the tightest lower bound (big omega). So, answer is $n \log n$.

Tightest lower bound of sorting (say S(n)) is $n \log n$ means there is no function f which has an order of growth larger than $n \log n$ and $f(n) = \Omega(S(n))$ holds.

A usual mistake is to think worst case changes with lower and upper bounds, but that is not the case. Worst case is defined for the algorithm and it is always the input which causes the algorithm the maximum complexity.

Correct Answer: C

62 votes -- Arjun Suresh (330k points)

1.3.8 Asymptotic Notations: GATE CSE 2005 | Question: 37 top https://gateoverflow.in/1373

✔ Applying Masters theorem,

$T(n) = \Theta(n \log n)$

So, it cannot be $\Omega(n^2)$.

Hence, answer is **(C)**.

32 votes -- shreya ghosh (2.8k points)

1.3.9 Asymptotic Notations: GATE CSE 2008 | Question: 39 top https://gateoverflow.in/450

✔ $g(n) = n!$.

On expanding the factorial we get $g(n) = O(n^n)$:

$$n^n > n^{\log n}$$
$$n^n \quad > 2^n$$

This condition is violated by options A, B and C by first statements of each. Hence, they cannot be said to be TRUE.

Second statement of option D says that $g(n)$ is asymptotically biggest of all.

Answer is option **(D).**

34 votes -- Amar Vashishth (25.2k points)

1.3.10 Asymptotic Notations: GATE CSE 2011 | Question: 37 top https://gateoverflow.in/2139

✔ Answer is (A).

$n \log_2 n < n^{3/2}$ is quite straightforward as $n^{3/2} = n \times n^{1/2}$ and $\log n < n^{1/2}$ as logarithmic growth is smaller than exponential growth however small be the exponentiation factor.

Also, $n^{3/2} < n^{\log_2 n}$ and $n^{3/2} < 2^n$.

Now only $n^{\log_2 n}$ and 2^n need to be compared.

Taking log of both $(\log_2 n)^2$ and n,

$n > (\log_2 n)^2$

Hence, $2^n > n^{\log_2 n}$.

NOTE: We cannot compare two functions for asymptotic growth by taking log if they are giving constants after $\log$ operation.

45 votes -- Ankit Rokde (6.9k points)

1.3.11 Asymptotic Notations: GATE CSE 2012 | Question: 18 top https://gateoverflow.in/50

✔ Worst case complexity can never be lower than the average case complexity, but it can be higher. So, (C) is the answer.

$A(n) = O(W(n))$.

47 votes -- Arjun Suresh (330k points)

1.3.12 Asymptotic Notations: GATE CSE 2015 Set 3 | Question: 4 top

https://gateoverflow.in/8398

✔ Sum of the cubes of the first n natural numbers is given by $(n(n+1)/2)^2$ which is $\Theta(n^4)$. So, I, III and IV are correct. II is wrong.

$\therefore$ **(C)** is correct.

55 votes

-- Arjun Suresh (330k points)

1.3.13 Asymptotic Notations: GATE CSE 2015 Set 3 | Question: 42 top

https://gateoverflow.in/8501

✔ **The answer is option (D).**

Since the value of $\sin(n)$ will always range from -1 to $+1$, hence $g(n)$ can take values 1, n, n^2.

Hence, if $g(n) = 1$, then statement I is incorrect.

And, if $g(n) = n^2$, then statement II is incorrect.

67 votes

-- saurabhrk (1.1k points)

1.3.14 Asymptotic Notations: GATE CSE 2017 Set 1 | Question: 04 top

https://gateoverflow.in/118703

✔ 10 is constant. $\therefore$ Growth rate is 0.

$\sqrt{n}$ grows slower than linear but faster than $\log$. (Consider $\frac{\sqrt{n^2}}{\sqrt{n}} = \sqrt{n}$, whereas $\frac{\log(n^2)}{\log n} = 2$)

n : Growth rate is linear.

$\log_2 n$: Growth rate is logarithmic. For asymptotic growth, the base does not matter.

$\frac{100}{n}$: Growth rate decreases with n.

So, correct answer is **(B)**.

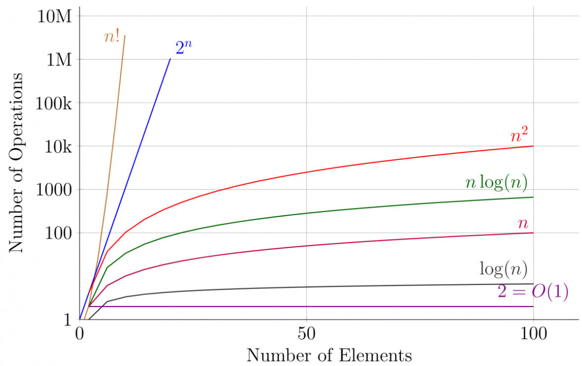
NOTE: Please never substitute large values of n in such questions. If ever you do, at least do for 2 such values and take ratio to get the growth rate or plot a graph. Remember $1.01^n \neq O\left(n^{100}\right)$.

49 votes

-- Arjun Suresh (330k points)

- 10, Constant
- $\sqrt{n}$, Square root
- n, Polynomial
- $\log_2 n$, Logarithmic
- $\frac{100}{n}$. Constant division by polynomial **(clearly less than constant for every value of n >100)**

Now we know Constant division by polynomial < Constant < Logarithmic <Square root < Polynomial



So, correct order is $\frac{100}{n}$, 10, $\log_2 n$, $\sqrt{n}$, n

PS: In the above graph, for asymptotic notations any graph can be multiplied by a constant and so the **SLOPE** is the important factor. If any plot is having higher slope, it'll eventually overtake the lower slope one at some value of x and hence asymptotically larger than the other.

34 votes -- Prashant Singh (47.1k points)

1.3.15 Asymptotic Notations: GATE CSE 2021 Set 1 | Question: 3 top https://gateoverflow.in/357449

Lets identify the function types

- $f_1 = 10^n -$ Exponential Function
- $f_2 = n^{\log n} -$ Super-polynomial but sub-exponential
- $f_3 = n^{\sqrt{n}} -$ Super-polynomial but sub-exponential

So, clearly asymptotic growth of f_1 is the highest.

Both f_2 and f_3 are super polynomial (grows faster than any polynomial function) but sub-exponential (grows slower than any exponential function).

Now, $\log n$ grows slower than any power function for positive power (not necessarily a polynomial function). i.e., $\log n = o(n^x), x > 0$. So, asymptotic growth of $\log n$ is lower than even $n^{0.000001}$. $\sqrt{n}$ is a power function with power $= 0.5$. So, clearly $\log n = o(\sqrt{n}) \implies n^{\log n} = o\left(n^{\sqrt{n}}\right)$.

So, asymptotic growth rate of f_1, f_2 and f_3 are in the order $f_2 < f_3 < f_1$.

Option D is correct.

2 votes -- gatecse (62.6k points)

1.3.16 Asymptotic Notations: GATE IT 2004 | Question: 55 top https://gateoverflow.in/3698

Answer is (D).

We can verify as : $f \leqslant g$ but $g \nleqslant f$. Therefore, $f < g$.

Also, $g = h$, as $g = O(h)$ and $h = O(g)$.

29 votes -- Sandeep_Uniyal (6.5k points)

1.3.17 Asymptotic Notations: GATE IT 2008 | Question: 10 top https://gateoverflow.in/3270

$A < C$ and $A < D$ are straight forward.

$E < B$ and $C, D < E$ as E and B are exponential functions and B having a larger base.

Now, we just need to see if C or D is larger.

In C we have a term $n^{3/4}$ and correspondingly in D we have $\log^9 n$ (after taking n out).

$n^{3/4}$ is asymptotically larger than $\log^9 n$ as when $n = 10^{100}, \log^9 n$ gives 100^9, while $n^{3/4}$ gives $10^{75} > 100^{37}$ a much higher value and this is true for all higher values of n. So, D < C.

Thus, A is correct.

50 votes -- Arjun Suresh (330k points)

1.4 Binary Search (1) top

1.4.1 Binary Search: GATE CSE 2021 Set 2 | Question: 8 top https://gateoverflow.in/357532

What is the worst-case number of arithmetic operations performed by recursive binary search on a sorted array of size n?

A. $\Theta(\sqrt{n})$
B. $\Theta(\log_2(n))$
C. $\Theta(n^2)$
D. $\Theta(n)$

gate2021-cse-set2 algorithms binary-search time-complexity

Answer

Answers: Binary Search

1.4.1 Binary Search: GATE CSE 2021 Set 2 | Question: 8 top https://gateoverflow.in/357532

✔ Worst-case number of arithmetic operations performed by recursive binary search on a sorted array is given by the following recurrence relation:

$T(n) = T(\frac{n}{2}) + \Theta(1)$

for $T(n) = aT(\frac{n}{b}) + f(n), a \geq 1, b > 1$

Case 2 of master theorem says,

if $f(n) = \Theta(n^{\log_b a})$then $T(n) = \Theta(n^{\log_b a} \cdot \log(n))$

Here, $n^{\log_b(a)} = n^{\log_2(1)} = n^0 = 1 \Rightarrow f(n) = \Theta(n^{\log_b(a)})$

Hence, we can conclude that $T(n) = \Theta(1 \cdot \log_2(n)) = \Theta(\log_2(n))$

B is correct

2 votes -- Nikhil Dhama (2.4k points)

1.5 Dynamic Programming (12) top

1.5.1 Dynamic Programming: GATE CSE 2008 | Question: 80 top https://gateoverflow.in/498

The subset-sum problem is defined as follows. Given a set of n positive integers, $S = \{a_1, a_2, a_3, \dots, a_n\}$, and positive integer W, is there a subset of S whose elements sum to W? A dynamic program for solving this problem uses a 2-dimensional Boolean array, X, with n rows and $W+1$ columns. $X[i,j], 1 \leq i \leq n, 0 \leq j \leq W$, is TRUE, if and only if there is a subset of $\{a_1, a_2, \dots, a_i\}$ whose elements sum to j.

Which of the following is valid for $2 \leq i \leq n$, and $a_i \leq j \leq W$?

A. $X[i,j] = X[i-1,j] \vee X[i,j-a_i]$
B. $X[i,j] = X[i-1,j] \vee X[i-1,j-a_i]$
C. $X[i,j] = X[i-1,j] \wedge X[i,j-a_i]$
D. $X[i,j] = X[i-1,j] \wedge X[i-1,j-a_i]$

gate2008-cse algorithms normal dynamic-programming

Answer

1.5.2 Dynamic Programming: GATE CSE 2008 | Question: 81 top https://gateoverflow.in/43484

The subset-sum problem is defined as follows. Given a set of n positive integers, $S = \{a_1, a_2, a_3, \dots, a_n\}$, and positive integer W, is there a subset of S whose elements sum to W? A dynamic program for solving this problem uses a 2-dimensional Boolean array, X, with n rows and $W+1$ columns. $X[i,j], 1 \leq i \leq n, 0 \leq j \leq W$, is TRUE, if and only if there is a subset of $\{a_1, a_2, \dots, a_i\}$ whose elements sum to j.

Which entry of the array X, if TRUE, implies that there is a subset whose elements sum to W?

A. $X[1,W]$
B. $X[n,0]$
C. $X[n,W]$
D. $X[n-1,n]$

gate2008-cse algorithms normal dynamic-programming

Answer

1.5.3 Dynamic Programming: GATE CSE 2009 | Question: 53 top https://gateoverflow.in/1338

A sub-sequence of a given sequence is just the given sequence with some elements (possibly none or all) left out. We are given two sequences $X[m]$ and $Y[n]$ of lengths m and n, respectively with indexes of X and Y starting from 0.

We wish to find the length of the longest common sub-sequence (LCS) of $X[m]$ and $Y[n]$ as $l(m,n)$, where an incomplete recursive definition for the function $I(i,j)$ to compute the length of the LCS of $X[m]$ and $Y[n]$ is given below:

```
l(i,j)  = 0, if either i = 0 or j = 0
        = expr1, if i,j > 0 and X[i-1] = Y[j-1]
        = expr2, if i,j > 0 and X[i-1] ≠ Y[j-1]
```

Which one of the following options is correct?

A. $\text{expr1} = l\,(i-1, j) + 1$
B. $\text{expr1} = l\,(i, j-1)$
C. $\text{expr2} = \max\,(l\,(i-1, j)\,, l\,(i, j-1))$
D. $\text{expr2} = \max\,(l\,(i-1, j-1)\,, l\,(i, j))$

gate2009-cse algorithms normal dynamic-programming recursion

Answer

1.5.4 Dynamic Programming: GATE CSE 2009 | Question: 54 top https://gateoverflow.in/43476

A sub-sequence of a given sequence is just the given sequence with some elements (possibly none or all) left out. We are given two sequences $X[m]$ and $Y[n]$ of lengths m and n, respectively with indexes of X and Y starting from 0.

We wish to find the length of the longest common sub-sequence (LCS) of $X[m]$ and $Y[n]$ as $l(m, n)$, where an incomplete recursive definition for the function $I(i, j)$ to compute the length of the LCS of $X[m]$ and $Y[n]$ is given below:

$l(i, j) = 0$, if either $i = 0$ or $j = 0$
$= expr1, \text{ if } i, j > 0 \text{ and } X[i-1] = Y[j-1]$
$= expr2, \text{ if } i, j > 0 \text{ and } X[i-1] \neq Y[j-1]$

The value of $l(i, j)$ could be obtained by dynamic programming based on the correct recursive definition of $l(i, j)$ of the form given above, using an array $L[M, N]$, where $M = m + 1$ and $N = n + 1$, such that $L[i, j] = l(i, j)$.

Which one of the following statements would be TRUE regarding the dynamic programming solution for the recursive definition of $l(i, j)$?

A. All elements of L should be initialized to 0 for the values of $l(i, j)$ to be properly computed.
B. The values of $l(i, j)$ may be computed in a row major order or column major order of $L[M, N]$.
C. The values of $l(i, j)$ cannot be computed in either row major order or column major order of $L[M, N]$.
D. $L[p, q]$ needs to be computed before $L[r, s]$ if either $p < r$ or $q < s$.

gate2009-cse normal algorithms dynamic-programming recursion

Answer

1.5.5 Dynamic Programming: GATE CSE 2010 | Question: 34 top https://gateoverflow.in/2208

The weight of a sequence $a_0, a_1, \dots, a_{n-1}$ of real numbers is defined as $a_0 + a_1/2 + \dots + a_{n-1}/2^{n-1}$. A subsequence of a sequence is obtained by deleting some elements from the sequence, keeping the order of the remaining elements the same. Let X denote the maximum possible weight of a subsequence of $a_o, a_1, \dots, a_{n-1}$ and Y the maximum possible weight of a subsequence of $a_1, a_2, \dots, a_{n-1}$. Then X is equal to

A. $max(Y, a_0 + Y)$
B. $max(Y, a_0 + Y/2)$
C. $max(Y, a_0 + 2Y)$
D. $a_0 + Y/2$

gate2010-cse algorithms dynamic-programming normal

Answer

1.5.6 Dynamic Programming: GATE CSE 2011 | Question: 25 top https://gateoverflow.in/2127

An algorithm to find the length of the longest monotonically increasing sequence of numbers in an array $A[0 : n-1]$ is given below.

Let L_i, denote the length of the longest monotonically increasing sequence starting at index i in the array.

Initialize $L_{n-1} = 1$.

For all i such that $0 \leq i \leq n-2$

$$L_i = \begin{cases} 1 + L_{i+1} & \text{if A[i] < A[i+1]} \\ 1 & \text{Otherwise} \end{cases}$$

Finally, the length of the longest monotonically increasing sequence is $\max\,(L_0, L_1, \dots, L_{n-1})$.

Which of the following statements is **TRUE**?

A. The algorithm uses dynamic programming paradigm

B. The algorithm has a linear complexity and uses branch and bound paradigm
C. The algorithm has a non-linear polynomial complexity and uses branch and bound paradigm
D. The algorithm uses divide and conquer paradigm

gate2011-cse algorithms easy dynamic-programming

Answer

1.5.7 Dynamic Programming: GATE CSE 2011 | Question: 38 top https://gateoverflow.in/2140

Four Matrices M_1, M_2, M_3 and M_4 of dimensions $p \times q$, $q \times r$, $r \times s$ and $s \times t$ respectively can be multiplied in several ways with different number of total scalar multiplications. For example when multiplied as $((M_1 \times M_2) \times (M_3 \times M_4))$, the total number of scalar multiplications is $pqr + rst + prt$. When multiplied as $(((M_1 \times M_2) \times M_3) \times M_4)$, the total number of scalar multiplications is $pqr + prs + pst$.

If $p = 10, q = 100, r = 20, s = 5$ and $t = 80$, then the minimum number of scalar multiplications needed is

A. 248000
B. 44000
C. 19000
D. 25000

gate2011-cse algorithms dynamic-programming normal

Answer

1.5.8 Dynamic Programming: GATE CSE 2014 Set 2 | Question: 37 top https://gateoverflow.in/1996

Consider two strings A=*"qpqrr"* and B=*"pqprqrp"*. Let x be the length of the longest common subsequence *(not necessarily contiguous)* between A and B and let y be the number of such longest common subsequences between A and B. Then $x + 10y =$ ___.

gate2014-cse-set2 algorithms normal numerical-answers dynamic-programming

Answer

1.5.9 Dynamic Programming: GATE CSE 2014 Set 3 | Question: 37 top https://gateoverflow.in/2071

Suppose you want to move from 0 to 100 on the number line. In each step, you either move right by a unit distance or you take a *shortcut*. A shortcut is simply a pre-specified pair of integers $i,\ j$ with $i < j$. Given a shortcut (i, j), if you are at position i on the number line, you may directly move to j. Suppose $T(k)$ denotes the smallest number of steps needed to move from k to 100. Suppose further that there is at most 1 shortcut involving any number, and in particular, from 9 there is a shortcut to 15. Let y and z be such that $T(9) = 1 + \min(T(y), T(z))$. Then the value of the product yz is _____.

gate2014-cse-set3 algorithms normal numerical-answers dynamic-programming

Answer

1.5.10 Dynamic Programming: GATE CSE 2016 Set 2 | Question: 14 top https://gateoverflow.in/39570

The Floyd-Warshall algorithm for all-pair shortest paths computation is based on

A. Greedy paradigm.
B. Divide-and-conquer paradigm.
C. Dynamic Programming paradigm.
D. Neither Greedy nor Divide-and-Conquer nor Dynamic Programming paradigm.

gate2016-cse-set2 algorithms dynamic-programming easy

Answer

1.5.11 Dynamic Programming: GATE CSE 2016 Set 2 | Question: 38 top https://gateoverflow.in/39587

Let A_1, A_2, A_3 and A_4 be four matrices of dimensions $10 \times 5, 5 \times 20, 20 \times 10$ and 10×5, respectively. The minimum number of scalar multiplications required to find the product $A_1 A_2 A_3 A_4$ using the basic matrix multiplication method is _________.

gate2016-cse-set2 dynamic-programming algorithms normal numerical-answers

Answer

1.5.12 Dynamic Programming: GATE CSE 2018 | Question: 31 top https://gateoverflow.in/204105

Assume that multiplying a matrix G_1 of dimension $p \times q$ with another matrix G_2 of dimension $q \times r$ requires pqr scalar multiplications. Computing the product of n matrices $G_1 G_2 G_3 \dots G_n$ can be done by parenthesizing in different ways. Define $G_i G_{i+1}$ as an **explicitly computed pair** for a given paranthesization if they are directly multiplied. For example, in the matrix multiplication chain $G_1 G_2 G_3 G_4 G_5 G_6$ using parenthesization $(G_1(G_2 G_3))(G_4(G_5 G_6))$, $G_2 G_3$ and $G_5 G_6$ are only explicitly computed pairs.

Consider a matrix multiplication chain $F_1 F_2 F_3 F_4 F_5$, where matrices F_1, F_2, F_3, F_4 and F_5 are of dimensions $2 \times 25, 25 \times 3, 3 \times 16, 16 \times 1$ and 1×1000, respectively. In the parenthesization of $F_1 F_2 F_3 F_4 F_5$ that minimizes the total number of scalar multiplications, the explicitly computed pairs is/are

A. $F_1 F_2$ and $F_3 F_4$ only
B. $F_2 F_3$ only
C. $F_3 F_4$ only
D. $F_1 F_2$ and $F_4 F_5$ only

gate2018-cse algorithms dynamic-programming

Answer

Answers: Dynamic Programming

1.5.1 Dynamic Programming: GATE CSE 2008 | Question: 80 top https://gateoverflow.in/498

✔ This is analogous to the dynamic programming solution to $0/1$ knapsack problem.

Consider the capacity of the knapsack, i.e., W to be analogous to J(the total sum here).

The solution **exploits the optimal substructure** of the problem.

At each stage we can have 2 options:

Case (1): Either we take an item(in this question either we consider the element A_i) along with the total solution to previous sub-problem(total solution here means the total sum obtained till previous sub-problem)

in which case we choose $A[i-1][j-a_i]$

$A[i-1]$ indicates we are considering solution to previous subproblem and

$A[j-a_i]$ means we have considered element a_i and now remaining sum is $J - a_i$ which has to be further considered.

Case (2): Or we do not consider the item(in this case the element a_i) in which case we only consider the solution to previous subproblem, which is, $A[i-1][J]$

Since the whole solution to this subset-sum problem is Logical OR(+) of cases 1 and 2, we eliminate options C and D because both are considering the Logical AND of the two parts of the solution.

Now, since here in the given question we are given a boolean array $X[n][W+1]$

So, an entry $X[i][j]$ is true only if sum j is possible with array elements from 0 to i.

So, for Each element of array, a_i, we consider two possibilities:

(1) Either we can ignore it and still get our possible sum, which is, $X[i-1][j]$

OR

(2) We could include element a_i and then get our required sum, which is, $X[i-1][j-a_i]$

And finally, to get $X[i][j]$, we take logical or of the above two cases.

Hence, **answer is option B.**

Reference :

Video:

By using the analogy of the problem and solution between subset-sum problem and $0/1$ knapsack problem, the above video clearly explains the how the solution to the problem is structured .

Video:

👍 23 votes

-- Ayush Upadhyaya (28.3k points)

1.5.2 Dynamic Programming: GATE CSE 2008 | Question: 81 top

☛ https://gateoverflow.in/43484

✔ **ANSWER is C.**

If LAST ROW and LAST COLUMN entry is 1, then there exists a subset whose elements sum to W.

👍 22 votes

-- Shivam Bhardwaj (263 points)

1.5.3 Dynamic Programming: GATE CSE 2009 | Question: 53 top

☛ https://gateoverflow.in/1338

✔ Answer is **C.** When the currently compared elements doesn't match, we have two possibilities for the LCS, one including X[i] but not Y[j] and other including Y[j] but not X[i].

```
/* Returns length of LCS for X[0..m-1], Y[0..n-1] */
int lcs( char *X, char *Y, int m, int n )
{
   if (m == 0 || n == 0)
     return 0;
   if (X[m-1] == Y[n-1])
     return 1 + lcs(X, Y, m-1, n-1);
   else
     return max(lcs(X, Y, m, n-1), lcs(X, Y, m-1, n));
}
```

👍 32 votes

-- Sona Praneeth Akula (3.4k points)

1.5.4 Dynamic Programming: GATE CSE 2009 | Question: 54 top

☛ https://gateoverflow.in/43476

✔ $\text{expr2} = \max\left(l\left(i-1,j\right), l\left(i,j-1\right)\right)$

When the currently compared elements doesn't match, we have two possibilities for the LCS, one including $X[i]$ but not $Y[j]$ and other including $Y[j]$ but not $X[i]$.

```
/* Returns length of LCS for X[0..m-1], Y[0..n-1] */
int lcs( char *X, char *Y, int m, int n )
{
   if (m == 0 || n == 0)
     return 0;
   if (X[m-1] == Y[n-1])
     return 1 + lcs(X, Y, m-1, n-1);
   else
     return max(lcs(X, Y, m, n-1), lcs(X, Y, m-1, n));
}
```

Answer is **B.** Dynamic programming is used to save the previously found LCS. So, for any index [p,q] all smaller ones should have been computed earlier. Option D is not correct as the condition given requires even L[3,2] to be computed before L[2,4] which is not a necessity if we follow row-major order.

```
int lcs( char *X, char *Y, int m, int n )
{
   int L[m+1][n+1];
   int i, j;

   /* Following steps build L[m+1][n+1] in bottom up fashion. Note
      that L[i][j] contains length of LCS of X[0..i-1] and Y[0..j-1] */
   for (i=0; i<=m; i++)
   {
     for (j=0; j<=n; j++)
     {
       if (i == 0 || j == 0)
         L[i][j] = 0;

       else if (X[i-1] == Y[j-1])
         L[i][j] = L[i-1][j-1] + 1;

       else
         L[i][j] = max(L[i-1][j], L[i][j-1]);
     }
   }

   /* L[m][n] contains length of LCS for X[0..n-1] and Y[0..m-1] */
```

```
    return L[m][n];
}
```

37 votes

-- Arjun Suresh (330k points)

1.5.5 Dynamic Programming: GATE CSE 2010 | Question: 34 top

https://gateoverflow.in/2208

$$S = \langle a_0, S_1 \rangle$$
$$S_1 = \langle a_1, a_2, a_3 \ldots a_{n-1} \rangle$$

Two possible cases arise:

1. **$\mathbf{a_0}$ is included in the max weight subsequence of $\mathbf{S}$:**

   In this case, $X = \text{weight}\left(\left(\langle a_0, S_1 \rangle\right)\right) = a_0 + \frac{Y}{2}$

2. **$\mathbf{a_0}$ is not included in the max weight subsequence of $\mathbf{S}$:**

   In this case, $X = \text{weight}(S_1) = Y$

Since the value of a_0 can be anything (negative or $< \frac{Y}{2}$ in general) $\{\because a_i \in \mathbb{R}\}$, it is possible that $Y > a_0 + \frac{Y}{2}$.
The maximum possible weight of a subsequence of S is given by:

$$X = \max\left(Y, a_0 + \frac{Y}{2}\right)$$

Thus, option B is correct.

92 votes

-- Pragy Agarwal (18.3k points)

1.5.6 Dynamic Programming: GATE CSE 2011 | Question: 25 top

https://gateoverflow.in/2127

(A) is the answer.

The algorithm is storing the optimal solutions to subproblems at each point (for each i), and then using it to derive the optimal solution of a bigger problem. And that is dynamic programming approach. And the program has linear time complexity.

http://stackoverflow.com/questions/1065433/what-is-dynamic-programming

Now, branch and bound comes when we explore all possible solutions (branch) and we backtrack as soon as we realise we won't get a solution (in classical backtracking we will retreat only when we won't find the solution). In backtracking : In each step, you check if this step satisfies all the conditions.
If it does : you continue generating subsequent solutions
If not : you go one step backward to check for another path

So, backtracking gives all possible solutions while branch and bound will give only the optimal one.

The given algorithm here is neither backtracking nor branch and bound. Because we are not branching anywhere in the solution space.

And the algorithm is not divide and conquer as we are not dividing the problem and then merging the solution as in the case of merge sort (where merge is the conquer step).

https://en.wikipedia.org/wiki/Divide_and_conquer_algorithms

References

60 votes

-- Arjun Suresh (330k points)

1.5.7 Dynamic Programming: GATE CSE 2011 | Question: 38 top

https://gateoverflow.in/2140

Answer is **C.**

Ordering:

- First Multiply $M_2 \times M_3$.
  This requires $100 * 20 * 5$ multiplications.
- Then Multiply $M_1 \times (M_2 \times M_3)$.
  This requires $10 * 100 * 5$ multiplications.
- Then Multiply $(M_1 \times (M_2 \times M_3)) \times M_4$.
  This requires $10 * 5 * 8$ multiplications.

Total 19000 Multiplications.

Brute Force approach - anyone can do.

No. of possible ordering for 4 matrices is C_3 where C_3 is the 3^{rd} Catalan number and given by $n = 3$ in $\frac{1}{n+1}{}^{2n}C_n = 5$.

So, here we have

1. $(M_1 \times M_2) \times (M_3 \times M_4)$
2. $(M_1 \times (M_2 \times M_3)) \times M_4$
3. $((M_1 \times M_2) \times M_3) \times M_4$
4. $M_1 \times (M_2 \times (M_3 \times M_4))$
5. $M_1 \times ((M_2 \times M_3) \times M_4))$

Each of these would give no. of multiplications required as

1. $pqr + rst + prt$
2. $qrs + pqs + pst$
3. $pqr + prs + pst$
4. $rst + qrt + pqt$
5. $qrs + qst + pst$

The last 2 are having qt terms which are the highest terms by far and hence we can avoid them from consideration $qt = 8000$ multiplied by one other term would be larger than any value in choice. So, just find the value of first 3 terms.

1. $pqr + rst + prt = 20000 + 8000 + 16000 = 44000$
2. $qrs + pqs + pst = 10000 + 5000 + 4000 = 19000$ - smallest value in choice, we can stop here.
3. $pqr + prs + pst$

Dynamic Programming Solution (should know Matrix Chain Ordering algorithm)

Here we have a chain of length 4.

Dynamic programming solution of Matrix chain ordering has the solution

$$m[i,j] = \begin{cases} 0 & \text{if } i = j \\ \min\limits_{i \le k < j} m[i][k] + m[k+1][j] + p_{i-1}p_jp_k & \text{if } i < j \end{cases}$$

So, we can fill the following table starting with the diagonals and moving upward diagonally. Here, $k < j$ but $\geq i, m[i,i] = 0$.

$p_0 = p = 10, p_1 = q = 100, p_2 = r = 20, p_3 = s = 5, p_4 = t = 80$.

	j=1	**j=2**	**j=3**	**j=4**
i=1	0	$p_0p_1p_2$ $= 20000$	$\min(m[1,1] + m[2,3] + p_0p_1p_3,$ $m[1,2] + m[3,3] + p_0p_2p_3)$ $= 15000$	$\min(m[1,1] + m[2,4] + p_0p_1p_4,$ $m[1,2] + m[3,4] + p_0p_2p_4,$ $m[1,3] + m[4,4] + p_0p_3p_4)$ $= 19000$
i=2		0	$p_1p_2p_3 = 10000$	$\min(m[2,2] + m[3,4] + p_1p_2p_4,$ $m[2][3] + m[4,4] + p_1p_3p_4)$ $= \min(160000, 50000) = 50000$
i=3			0	$p_2p_3p_4 = 8000$
i=4				0

Our required answer is given by $m[1,4] = 19000$.

References

35 votes -- Sona Praneeth Akula (3.4k points)

1.5.8 Dynamic Programming: GATE CSE 2014 Set 2 | Question: 37 top https://gateoverflow.in/1996

Answer is 34.

In first string, if we want to get 4 as maximum length then LCS should end with either "rr" or "qr".
Only 4 combinations are possible for LCS with length 4:

"$qpqr$", "$qqrr$","$pqrr$","$qprr$"

Now, check for matching sequences in second string, except for "$qqrr$" all are possible.

34 votes -- Anurag Semwal (6.7k points)

1.5.9 Dynamic Programming: GATE CSE 2014 Set 3 | Question: 37 top https://gateoverflow.in/2071

$T(k)$ is the smallest number of steps needed to move from k to 100.

Now, it is given that y and z are two numbers such that,

$T(9) = 1 + \min(T(y), T(z))$, i.e.,

$T(9) = 1 + \min($Steps from y to 100, Steps from z to 100$)$, where y and z are two possible values that can be reached from 9.

One number that can be reached from 9 is 10, which is the number obtained if we simply move one position right on the number line. Another number is 15, the shortcut path from 9, as given in the question. So, we have two paths from 9, one is 10 and the other is 15.

Therefore, the value of y and z is 10 and 15 (either variable may take either of the values).

Thus, $yz = 150$.

53 votes -- Divya Bharti (8.8k points)

$T(9) =$ Distance from 9 to 100
$T(9) = 1 + \min(T(y), T(z)) =$ 1+min(Distance from y to 100 , Distance from z to 100)

There are only two such values where we can reach from 9 , one is simple step to right on number line , i.e 10 and another is 15 (given shortcut)

Hence , $y = 10$, $z = 15$
$yz = 10 \times 15 = 150$

36 votes -- Srinath Jayachandran (2.9k points)

1.5.10 Dynamic Programming: GATE CSE 2016 Set 2 | Question: 14 top https://gateoverflow.in/39570

In Floyd Warshall's, we calculate all possibilities and select best one so its neither Divide & Conquer nor Greedy but based on Dynamic Programming Paradigm.

Correct Answer: C

31 votes -- Anurag Semwal (6.7k points)

1.5.11 Dynamic Programming: GATE CSE 2016 Set 2 | Question: 38 top https://gateoverflow.in/39587

The answer is 1500.

Matrix Parenthesizing : $A_1((A_2A_3)A_4)$

Check my solution below, using dynamic programming

A_1	A_2	A_3	A_4
10×5	5×20	20×10	10×5

- $A_{12} = 10 \times 5 \times 20 = 1000$
- $A_{23} = 5 \times 20 \times 10 = 1000$
- $A_{34} = 20 \times 10 \times 5 = 1000$

$$A_{13} = min \begin{cases} A_{12} + A_{33} + 5 \times 20 \times 10 = 2000 \\ A_{11} + A_{23} + 10 \times 5 \times 10 = 1500 \end{cases}$$

$$A_{24} = min \begin{cases} A_{23} + A_{44} + 5 \times 10 \times 5 = 1250 \\ A_{22} + A_{34} + 5 \times 20 \times 5 = 1500 \end{cases}$$

$$A_{14} = min \begin{cases} A_{11} + A_{24} + 10 \times 5 \times 5 = 1500 \\ A_{12} + A_{34} + 10 \times 20 \times 5 \geqslant 2000 \\ A_{13} + A_{44} + 10 \times 20 \times 5 = 2000 \end{cases}$$

Answer is 1500.

44 votes -- Akash Kanase (36k points)

1.5.12 Dynamic Programming: GATE CSE 2018 | Question: 31 top https://gateoverflow.in/204105

✔ If we multiply anything with F_5 we will get much greater multiplication cost because F_5 is $1 * 1000$ matrix so 1000 will play vital role in cost. So we will multiply F_5 at very last step.

So, here is the sequence giving minimal cost:

$(F_1(F_2(F_3F_4)))(F_5) = 48 + 75 + 50 + 2000 = 2173$

Explicitly computed pairs is (F_3F_4)

Correct Answer: C

37 votes -- Digvijay (44.9k points)

1.6 Graph Algorithms (36) top

1.6.1 Graph Algorithms: GATE CSE 1994 | Question: 1.22 top https://gateoverflow.in/2465

Which of the following statements is false?

A. Optimal binary search tree construction can be performed efficiently using dynamic programming
B. Breadth-first search cannot be used to find connected components of a graph
C. Given the prefix and postfix walks over a binary tree, the binary tree cannot be uniquely constructed.
D. Depth-first search can be used to find connected components of a graph

gate1994 algorithms normal graph-algorithms

Answer

1.6.2 Graph Algorithms: GATE CSE 1994 | Question: 24 top https://gateoverflow.in/2520

An independent set in a graph is a subset of vertices such that no two vertices in the subset are connected by an edge. An incomplete scheme for a greedy algorithm to find a maximum independent set in a tree is given below:

```
V: Set of all vertices in the tree;
I := ϕ
while V ≠ ϕ do
begin
    select a vertex u ∊ V such that
    _______;
    V := V - {u};
    if u is such that
    ________then I := I ∪ {u}
end;
Output(I);
```

A. Complete the algorithm by specifying the property of vertex u in each case.
B. What is the time complexity of the algorithm?

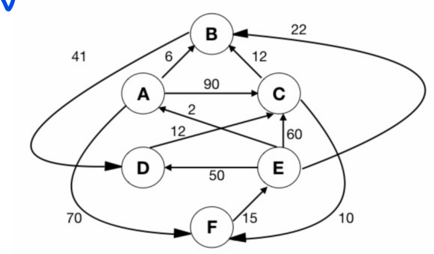
gate1994 algorithms graph-algorithms normal descriptive

Answer

1.6.3 Graph Algorithms: GATE CSE 1996 | Question: 17 top

https://gateoverflow.in/2769

Let G be the directed, weighted graph shown in below figure



We are interested in the shortest paths from A.

a. Output the sequence of vertices identified by the Dijkstra's algorithm for single source shortest path when the algorithm is started at node A
b. Write down sequence of vertices in the shortest path from A to E
c. What is the cost of the shortest path from A to E?

gate1996 algorithms graph-algorithms normal descriptive

Answer

1.6.4 Graph Algorithms: GATE CSE 1998 | Question: 1.21, ISRO2008-16 top

https://gateoverflow.in/1658

Which one of the following algorithm design techniques is used in finding all pairs of shortest distances in a graph?

A. Dynamic programming
B. Backtracking
C. Greedy
D. Divide and Conquer

gate1998 algorithms graph-algorithms easy isro2008

Answer

1.6.5 Graph Algorithms: GATE CSE 2002 | Question: 12 top

https://gateoverflow.in/865

Fill in the blanks in the following template of an algorithm to compute all pairs shortest path lengths in a directed graph G with $n * n$ adjacency matrix A. $A[i, j]$ equals 1 if there is an edge in G from i to j, and 0 otherwise. Your aim in filling in the blanks is to ensure that the algorithm is correct.

```
INITIALIZATION: For i = 1 ... n
    {For j = 1 ... n
        { if a[i,j] = 0 then P[i,j] =_______ else P[i,j] =_______;}
    }

ALGORITHM: For i = 1 ... n
    {For j = 1 ... n
        {For k = 1 ... n
            {P[__,__] = min{_______,______}; }
        }
    }
```

a. Copy the complete line containing the blanks in the Initialization step and fill in the blanks.
b. Copy the complete line containing the blanks in the Algorithm step and fill in the blanks.
c. Fill in the blank: The running time of the Algorithm is $O(\_\_\_)$.

gate2002-cse algorithms graph-algorithms time-complexity normal descriptive

Answer

1.6.6 Graph Algorithms: GATE CSE 2003 | Question: 67 top

https://gateoverflow.in/954

Let $G=(V,E)$ be an undirected graph with a subgraph $G_1=(V_1,E_1)$. Weights are assigned to edges of G as follows.

$$w(e)=\begin{cases}0, & \text{if } e \in E_1\\ 1, & \text{otherwise}\end{cases}$$

A single-source shortest path algorithm is executed on the weighted graph (V,E,w) with an arbitrary vertex v_1 of V_1 as the source. Which of the following can always be inferred from the path costs computed?

A. The number of edges in the shortest paths from v_1 to all vertices of G
B. G_1 is connected
C. V_1 forms a clique in G
D. G_1 is a tree

gate2003-cse algorithms graph-algorithms normal

Answer

1.6.7 Graph Algorithms: GATE CSE 2003 | Question: 70 top

https://gateoverflow.in/957

Let $G=(V,E)$ be a directed graph with n vertices. A path from v_i to v_j in G is a sequence of vertices ($v_i, v_{i+1}, \ldots, v_j$) such that $(v_k, v_{k+1}) \in E$ for all k in i through $j-1$. A simple path is a path in which no vertex appears more than once.

Let A be an $n \times n$ array initialized as follows:

$$A[j,k]=\begin{cases}1, & \text{if } (j,k) \in E\\ 0, & \text{otherwise}\end{cases}$$

Consider the following algorithm:

```
for i=1 to n
    for j=1 to n
        for k=1 to n
            A[j,k] = max(A[j,k], A[j,i] + A[i,k]);
```

Which of the following statements is necessarily true for all j and k after termination of the above algorithm?

A. $A[j,k] \leq n$
B. If $A[j,j] \geq n-1$ then G has a Hamiltonian cycle
C. If there exists a path from j to k, $A[j,k]$ contains the longest path length from j to k
D. If there exists a path from j to k, every simple path from j to k contains at most $A[j,k]$ edges

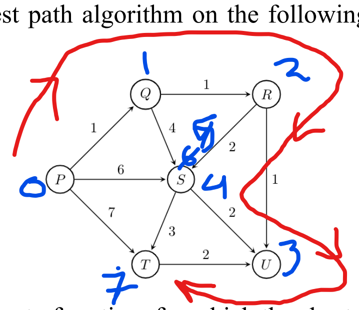
gate2003-cse algorithms graph-algorithms normal

Answer

1.6.8 Graph Algorithms: GATE CSE 2004 | Question: 44 top

https://gateoverflow.in/1041

Suppose we run Dijkstra's single source shortest path algorithm on the following edge-weighted directed graph with vertex P as the source.



In what order do the nodes get included into the set of vertices for which the shortest path distances are finalized?

A. P, Q, R, S, T, U
B. P, Q, R, U, S, T
C. P, Q, R, U, T, S
D. P, Q, T, R, U, S

gate2004-cse algorithms graph-algorithms normal

Answer

1.6.9 Graph Algorithms: GATE CSE 2004 | Question: 81 top

https://gateoverflow.in/1075

Let $G_1 = (V, E_1)$ and $G_2 = (V, E_2)$ be connected graphs on the same vertex set V with more than two vertices. If $G_1 \cap G_2 = (V, E_1 \cap E_2)$ is not a connected graph, then the graph $G_1 \cup G_2 = (V, E_1 \cup E_2)$

A. cannot have a cut vertex
B. must have a cycle
C. must have a cut-edge (bridge)
D. has chromatic number strictly greater than those of G_1 and G_2

gate2004-cse algorithms graph-algorithms normal

Answer

1.6.10 Graph Algorithms: GATE CSE 2005 | Question: 38 top

https://gateoverflow.in/1374

Let $G(V, E)$ be an undirected graph with positive edge weights. Dijkstra's single source shortest path algorithm can be implemented using the binary heap data structure with time complexity:

A. $O\left(|V|^2\right)$
B. $O(|E| + |V| \log |V|)$
C. $O(|V| \log |V|)$
D. $O((|E| + |V|) \log |V|)$

gate2005-cse algorithms graph-algorithms normal

Answer

1.6.11 Graph Algorithms: GATE CSE 2005 | Question: 82a top

https://gateoverflow.in/1404

Let s and t be two vertices in a undirected graph $G = (V, E)$ having distinct positive edge weights. Let $[X, Y]$ be a partition of V such that $s \in X$ and $t \in Y$. Consider the edge e having the minimum weight amongst all those edges that have one vertex in X and one vertex in Y.

The edge e must definitely belong to:

A. the minimum weighted spanning tree of G
B. the weighted shortest path from s to t
C. each path from s to t
D. the weighted longest path from s to t

gate2005-cse algorithms graph-algorithms normal

Answer

1.6.12 Graph Algorithms: GATE CSE 2005 | Question: 82b top

https://gateoverflow.in/82129

Let s and t be two vertices in a undirected graph $G = (V, E)$ having distinct positive edge weights. Let $[X, Y]$ be a partition of V such that $s \in X$ and $t \in Y$. Consider the edge e having the minimum weight amongst all those edges that have one vertex in X and one vertex in Y.

Let the weight of an edge e denote the congestion on that edge. The congestion on a path is defined to be the maximum of the congestions on the edges of the path. We wish to find the path from s to t having minimum congestion. Which of the following paths is always such a path of minimum congestion?

A. a path from s to t in the minimum weighted spanning tree
B. a weighted shortest path from s to t
C. an Euler walk from s to t
D. a Hamiltonian path from s to t

gate2005-cse algorithms graph-algorithms normal

Answer

1.6.13 Graph Algorithms: GATE CSE 2006 | Question: 12 top

☛ https://gateoverflow.in/891

To implement Dijkstra's shortest path algorithm on unweighted graphs so that it runs in linear time, the data structure to be used is:

imp

A. Queue
B. Stack
C. Heap
D. B-Tree

gate2006-cse algorithms graph-algorithms easy

Answer

1.6.14 Graph Algorithms: GATE CSE 2006 | Question: 48 top

☛ https://gateoverflow.in/1824

Let T be a depth first search tree in an undirected graph G. Vertices u and ν are leaves of this tree T. The degrees of both u and ν in G are at least 2. which one of the following statements is true?

A. There must exist a vertex w adjacent to both u and ν in G
B. There must exist a vertex w whose removal disconnects u and ν in G
C. There must exist a cycle in G containing u and ν
D. There must exist a cycle in G containing u and all its neighbours in G

gate2006-cse algorithms graph-algorithms normal

Answer

1.6.15 Graph Algorithms: GATE CSE 2007 | Question: 41 top

☛ https://gateoverflow.in/1239

In an unweighted, undirected connected graph, the shortest path from a node S to every other node is computed most efficiently, in terms of *time complexity*, by

A. Dijkstra's algorithm starting from S.
B. Warshall's algorithm.
C. Performing a DFS starting from S.
D. Performing a BFS starting from S.

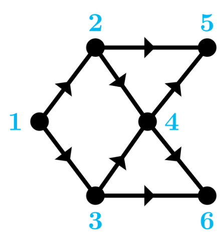
gate2007-cse algorithms graph-algorithms easy

Answer

1.6.16 Graph Algorithms: GATE CSE 2007 | Question: 5 top

☛ https://gateoverflow.in/31821

Consider the DAG with $V = \{1, 2, 3, 4, 5, 6\}$ shown below.



Which of the following is not a topological ordering?

A. 1 2 3 4 5 6
B. 1 3 2 4 5 6
C. 1 3 2 4 6 5
D. 3 2 4 1 6 5

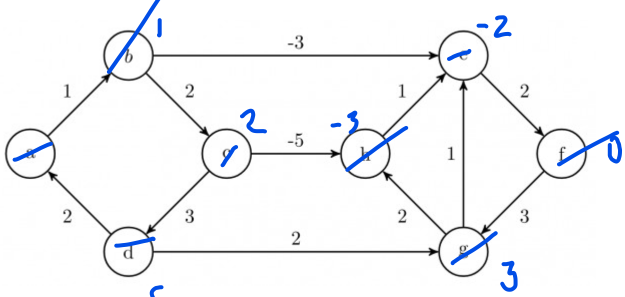
gate2007-cse algorithms graph-algorithms

Answer

1.6.17 Graph Algorithms: GATE CSE 2008 | Question: 45 top

https://gateoverflow.in/457

V.imp comments are present in the comment section



Dijkstra's single source shortest path algorithm when run from vertex a in the above graph, computes the correct shortest path distance to

A. only vertex a
B. only vertices a, e, f, g, h
C. only vertices a, b, c, d
D. all the vertices

gate2008-cse algorithms graph-algorithms normal

Answer

1.6.18 Graph Algorithms: GATE CSE 2008 | Question: 7 top

https://gateoverflow.in/405

The most efficient algorithm for finding the number of connected components in an undirected graph on n vertices and m edges has time complexity

A. $\Theta(n)$
B. $\Theta(m)$
C. $\Theta(m+n)$
D. $\Theta(mn)$

gate2008-cse algorithms graph-algorithms time-complexity normal

Answer

1.6.19 Graph Algorithms: GATE CSE 2009 | Question: 13 top

https://gateoverflow.in/1305

Which of the following statement(s) is/are correct regarding Bellman-Ford shortest path algorithm?

P: Always finds a negative weighted cycle, if one exists.

Q: Finds whether any negative weighted cycle is reachable from the source.

Always always use the BF method by RBR sir for BF algo.

A. P only
B. Q only
C. Both P and Q
D. Neither P nor Q

gate2009-cse algorithms graph-algorithms normal

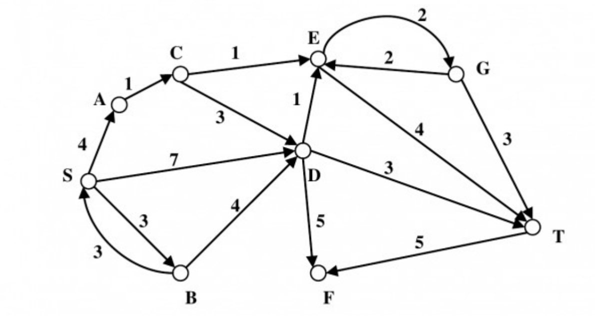
Answer

1.6.20 Graph Algorithms: GATE CSE 2012 | Question: 40 top

https://gateoverflow.in/1765

Consider the directed graph shown in the figure below. There are multiple shortest paths between vertices S and T. Which one will be reported by Dijkstra's shortest path algorithm? Assume that, in any iteration, the shortest path to a vertex v is updated only when a strictly shorter path to v is discovered.

Always always use the table method for Dijkstra.



A. SDT
B. SBDT
C. SACDT
D. SACET

gate2012-cse algorithms graph-algorithms normal

Answer

1.6.21 Graph Algorithms: GATE CSE 2013 | Question: 19 top

https://gateoverflow.in/1441

What is the time complexity of Bellman-Ford single-source shortest path algorithm on a complete graph of n vertices?

A. $\theta(n^2)$
B. $\theta(n^2 \log n)$
C. $\theta(n^3)$
D. $\theta(n^3 \log n)$

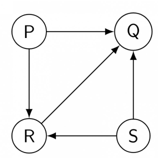
gate2013-cse algorithms graph-algorithms normal

Answer

1.6.22 Graph Algorithms: GATE CSE 2014 Set 1 | Question: 13 top

https://gateoverflow.in/1779

Consider the directed graph below given.



Which one of the following is **TRUE**?

A. The graph does not have any topological ordering.
B. Both PQRS and SRQP are topological orderings.
C. Both PSRQ and SPRQ are topological orderings.
D. PSRQ is the only topological ordering.

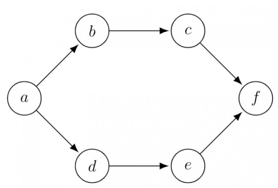
gate2014-cse-set1 graph-algorithms easy

Answer

1.6.23 Graph Algorithms: GATE CSE 2016 Set 1 | Question: 11 top

https://gateoverflow.in/39669

Consider the following directed graph:



The number of different topological orderings of the vertices of the graph is _____________.

gate2016-cse-set1 algorithms graph-algorithms normal numerical-answers

Answer

1.6.24 Graph Algorithms: GATE CSE 2016 Set 2 | Question: 41 top https://gateoverflow.in/39620

gg

In an adjacency list representation of an undirected simple graph $G = (V, E)$, each edge (u, v) has two adjacency list entries: $[v]$ in the adjacency list of u, and $[u]$ in the adjacency list of v. These are called twins of each other. A twin pointer is a pointer from an adjacency list entry to its twin. If $|E| = m$ and $|V| = n$, and the memory size is not a constraint, what is the time complexity of the most efficient algorithm to set the twin pointer in each entry in each adjacency list?

A. $\Theta\left(n^2\right)$
B. $\Theta\left(n+m\right)$
C. $\Theta\left(m^2\right)$
D. $\Theta\left(n^4\right)$

gate2016-cse-set2 algorithms graph-algorithms normal

Answer

1.6.25 Graph Algorithms: GATE CSE 2017 Set 1 | Question: 26 top https://gateoverflow.in/118306

Let $G = (V, E)$ be *any* connected, undirected, edge-weighted graph. The weights of the edges in E are positive and distinct. Consider the following statements:

I. Minimum Spanning Tree of G is always unique.
II. Shortest path between any two vertices of G is always unique.

Which of the above statements is/are necessarily true?

A. I only
B. II only
C. both I and II
D. neither I nor II

gate2017-cse-set1 algorithms graph-algorithms normal

Answer

1.6.26 Graph Algorithms: GATE CSE 2018 | Question: 43 top https://gateoverflow.in/204117

Let G be a graph with $100!$ vertices, with each vertex labelled by a distinct permutation of the numbers $1, 2, \dots, 100$. There is an edge between vertices u and v if and only if the label of u can be obtained by swapping two adjacent numbers in the label of v. Let y denote the degree of a vertex in G, and z denote the number of connected components in G. Then, $y + 10z =$ ____

gate2018-cse algorithms graph-algorithms numerical-answers

Answer

1.6.27 Graph Algorithms: GATE CSE 2020 | Question: 40 top https://gateoverflow.in/333191

Let $G = (V, E)$ be a directed, weighted graph with weight function $w : E \to \mathbb{R}$. For some function $f : V \to \mathbb{R}$, for each edge $(u, v) \in E$, define $w'(u, v)$ as $w(u, v) + f(u) - f(v)$.

Which one of the options completes the following sentence so that it is TRUE? Toughest and very nice question

"The shortest paths in G under w are shortest paths under w' too,____________".

A. for every $f : V \to \mathbb{R}$
B. if and only if $\forall u \in V,\ f(u)$ is positive
C. if and only if $\forall u \in V,\ f(u)$ is negative
D. if and only if $f(u)$ is the distance from s to u in the graph obtained by adding a new vertex s to G and edges of zero weight from s to every vertex of G

gate2020-cse algorithms graph-algorithms

Answer

1.6.28 Graph Algorithms: GATE CSE 2020 | Question: 49 top

https://gateoverflow.in/333182

Consider a graph $G = (V, E)$, where $V = \{v_1, v_2, \ldots, v_{100}\}$, $E = \{(v_i, v_j) \mid 1 \leq i < j \leq 100\}$, and weight of the edge (v_i, v_j) is $|i - j|$. The weight of minimum spanning tree of G is ________

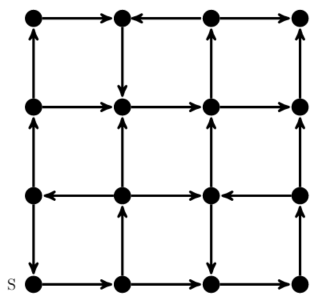
gate2020-cse numerical-answers algorithms graph-algorithms

Answer

1.6.29 Graph Algorithms: GATE CSE 2021 Set 2 | Question: 46 top

https://gateoverflow.in/357494

Consider the following directed graph:



Which of the following is/are correct about the graph?

A. The graph does not have a topological order
B. A depth-first traversal starting at vertex S classifies three directed edges as back edges
C. The graph does not have a strongly connected component
D. For each pair of vertices u and v, there is a directed path from u to v

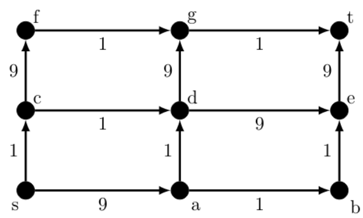
gate2021-cse-set2 multiple-selects algorithms graph-algorithms

Answer

1.6.30 Graph Algorithms: GATE CSE 2021 Set 2 | Question: 55 top

https://gateoverflow.in/357482

In a directed acyclic graph with a source vertex s, the *quality-score* of a directed path is defined to be the product of the weights of the edges on the path. Further, for a vertex v other than s, the quality-score of v is defined to be the maximum among the quality-scores of all the paths from s to v. The quality-score of s is assumed to be 1.



The sum of the quality-scores of all vertices on the graph shown above is _______

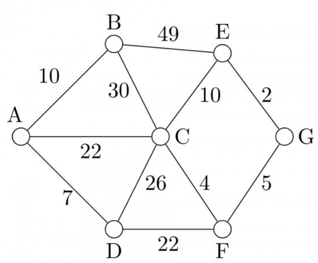
gate2021-cse-set2 algorithms graph-algorithms dag numerical-answers

Answer

1.6.31 Graph Algorithms: GATE IT 2004 | Question: 56 top

https://gateoverflow.in/3699

Consider the undirected graph below:



Using Prim's algorithm to construct a minimum spanning tree starting with node A, which one of the following sequences of edges represents a possible order in which the edges would be added to construct the minimum spanning tree?

A. (E, G), (C, F), (F, G), (A, D), (A, B), (A, C)
B. (A, D), (A, B), (A, C), (C, F), (G, E), (F, G)
C. (A, B), (A, D), (D, F), (F, G), (G, E), (F, C)
D. (A, D), (A, B), (D, F), (F, C), (F, G), (G, E)

gate2004-it algorithms graph-algorithms normal

Answer

1.6.32 Graph Algorithms: GATE IT 2005 | Question: 15 top https://gateoverflow.in/3760

In the following table, the left column contains the names of standard graph algorithms and the right column contains the time complexities of the algorithms. Match each algorithm with its time complexity.

1.	Bellman-Ford algorithm	A:	$O(m \log n)$
2.	Kruskal's algorithm	B:	$O(n^3)$
3.	Floyd-Warshall algorithm	C:	$O(nm)$
4.	Topological sorting	D:	$O(n+m)$

A. 1→ C, 2 → A, 3 → B, 4 → D
B. 1→ B, 2 → D, 3 → C, 4 → A
C. 1→ C, 2 → D, 3 → A, 4 → B
D. 1→ B, 2 → A, 3 → C, 4 → D

gate2005-it algorithms graph-algorithms normal

Answer

1.6.33 Graph Algorithms: GATE IT 2005 | Question: 84a top https://gateoverflow.in/3856

A sink in a directed graph is a vertex i such that there is an edge from every vertex $j \neq i$ to i and there is no edge from i to any other vertex. A directed graph G with n vertices is represented by its adjacency matrix A, where $A[i][j] = 1$ if there is an edge directed from vertex i to j and 0 otherwise. The following algorithm determines whether there is a sink in the graph G.

```
i = 0;
do {
    j = i + 1;
    while ((j < n) && E1) j++;
    if (j < n) E2;
} while (j < n);
flag = 1;
for (j = 0; j < n; j++)
    if ((j! = i) && E3) flag = 0;
if (flag) printf("Sink exists");
else printf ("Sink does not exist");
```

Choose the correct expressions for E_1 and E_2

A. $E_1 : A[i][j]$ and $E_2 : i = j;$
B. $E_1 :\ !A[i][j]$ and $E_2 : i = j + 1;$
C. $E_1 :\ !A[i][j]$ and $E_2 : i = j;$
D. $E_1 : A[i][j]$ and $E_2 : i = j + 1;$

gate2005-it algorithms graph-algorithms normal

Answer

1.6.34 Graph Algorithms: GATE IT 2005 | Question: 84b top https://gateoverflow.in/3857

A sink in a directed graph is a vertex i such that there is an edge from every vertex $j \neq i$ to i and there is no edge from i to any other vertex. A directed graph G with n vertices is represented by its adjacency matrix A, where $A[i][j] = 1$ if there is an edge directed from vertex i to j and 0 otherwise. The following algorithm determines whether there is a sink in the graph G.

```
i = 0;
do {
    j = i + 1;
    while ((j < n) && E1) j++;
    if (j < n) E2;
} while (j < n);
flag = 1;
for (j = 0; j < n; j++)
    if ((j! = i) && E3) flag = 0;
if (flag) printf("Sink exists") ;
else printf ("Sink does not exist");
```

Choose the correct expression for E_3

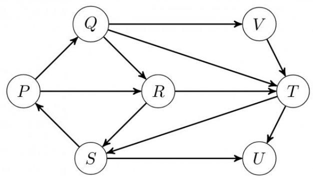
A. $(A[i][j]$ && $!A[j][i])$
B. $(!A[i][j]$ && $A[j][i])$
C. $(!A[i][j] \ || \ A[j][i])$
D. $(A[i][j] \ || \ !A[j][i])$

gate2005-it algorithms graph-algorithms normal

Answer

1.6.35 Graph Algorithms: GATE IT 2006 | Question: 46 top https://gateoverflow.in/3589

Which of the following is the correct decomposition of the directed graph given below into its strongly connected components?



A. $\{P, Q, R, S\}, \{T\}, \{U\}, \{V\}$
B. $\{P, Q, R, S, T, V\}, \{U\}$
C. $\{P, Q, S, T, V\}, \{R\}, \{U\}$
D. $\{P, Q, R, S, T, U, V\}$

gate2006-it algorithms graph-algorithms normal

Answer

1.6.36 Graph Algorithms: GATE IT 2007 | Question: 3, UGCNET-June2012-III: 34 top https://gateoverflow.in/3434

Consider a weighted, undirected graph with positive edge weights and let uv be an edge in the graph. It is known that the shortest path from the source vertex s to u has weight 53 and the shortest path from s to v has weight 65. Which one of the following statements is always TRUE?

A. Weight $(u, v) \leq 12$
B. Weight $(u, v) = 12$
C. Weight $(u, v) \geq 12$
D. Weight $(u, v) > 12$

gate2007-it algorithms graph-algorithms normal ugcnetjune2012iii

Answer

Answers: Graph Algorithms

1.6.1 Graph Algorithms: GATE CSE 1994 | Question: 1.22 top https://gateoverflow.in/2465

✔ The answer is B.

A. True.
B. False.
C. True.

D. True.

👍 19 votes -- Rajarshi Sarkar (27.8k points)

1.6.2 Graph Algorithms: GATE CSE 1994 | Question: 24 top ☛ https://gateoverflow.in/2520

✔

A. While adding vertex u to I it should not have an edge with any node in I.
B. The algorithm runs till V is empty (in $O(n)$ time) and is checking u with each vertex v in set I (in $O(n)$ time). So, overall complexity $O(n^2)$.

👍 24 votes -- Rajarshi Sarkar (27.8k points)

1.6.3 Graph Algorithms: GATE CSE 1996 | Question: 17 top ☛ https://gateoverflow.in/2769

✔

```
DIJKSTRA(G, w, s)
INITIALIZE-SINGLE-SOURCE(G, s)
S = ∅
Q = G.V
while Q ≠ ∅
    u = EXTRACT-MIN(Q)
    S = S ∪ {u}
    for each vertex v ∈ G.Adj[u]
        RELAX(u, v, w)
```

Correct Solutions:
(A).

$$Q = \left\{ \begin{array}{cccccc} \boxed{0} & \boxed{\infty} & \boxed{\infty} & \boxed{\infty} & \boxed{\infty} & \boxed{\infty} \\ \text{A}, & \text{B}, & \text{C}, & \text{D}, & \text{E}, & \text{F} \end{array} \right\}$$

- First we visit A as it is the one with smallest distance 0
- Relax operation updates distances to B,C,F as $6, 90, 70$ respectively

$$Q = \left\{ \begin{array}{ccccc} \boxed{6} & \boxed{90} & \boxed{\infty} & \boxed{\infty} & \boxed{70} \\ \text{B}, & \text{C}, & \text{D}, & \text{E}, & \text{F} \end{array} \right\}$$

- B is next visited as its distance is now 6
- Relax operation updates distance to D as $41 + 6 = 47$

$$Q = \left\{ \begin{array}{cccc} \boxed{90} & \boxed{47} & \boxed{\infty} & \boxed{70} \\ \text{C}, & \text{D}, & \text{E}, & \text{F} \end{array} \right\}$$

- D is next visited as its distance is now 47
- Relax operation updates distance to C as $47 + 12 = 59$

$$Q = \left\{ \begin{array}{ccc} \boxed{59} & \boxed{\infty} & \boxed{70} \\ \text{C}, & \text{E}, & \text{F} \end{array} \right\}$$

- C is next visited as its distance is now 59
- Relax operation updates distance to F as $59 + 10 = 69$

$$Q = \left\{ \begin{array}{cc} \boxed{\infty} & \boxed{69} \\ \text{E}, & \text{F} \end{array} \right\}$$

- F is next visited as its distance is now 69
- Relax operation updates distance to E as $69 + 15 = 84$

$$Q = \left\{ \begin{array}{c} \boxed{84} \\ \text{E} \end{array} \right\}$$

- Finally E is visited.

So, the sequence of node visits are A, B, D, C, F, E

(B). Sequence of vertices in the shortest path from A to E: $\text{A} - \text{B} - \text{D} - \text{C} - \text{F} - \text{E}$

(C). Cost of the shortest path from A to $\text{E} = 84$.

👍 28 votes -- Manu Thakur (34.1k points)

1.6.4 Graph Algorithms: GATE CSE 1998 | Question: 1.21, ISRO2008-16 top https://gateoverflow.in/1658

✔ Answer is (A) because Floyd Warshall algorithm is used to find all shortest paths which is a dynamic programming approach.

👍 22 votes -- shashi shekhar (437 points)

1.6.5 Graph Algorithms: GATE CSE 2002 | Question: 12 top https://gateoverflow.in/865

✔

```
INITIALIZATION:
For i = 1 ... n {
   For j = 1 ... n {
      if a[i,j] = 0 then P[i,j] = infinite
        // i.e. if there is no direct path then put path length as infinite
       else P[i,j] =a[i,j];
   }
}

ALGORITHM:
For i = 1 ... n { //i loops over the intermediate vertices
   For j = 1 ... n {
      For k = 1 ... n {
               P[j, k] = min( p[j,k] , p[j,i] + p[i,k]);
      }
   }
}
```

Time complexity $O(n^3)$

This algorithm is for weighted graph but it will work for unweighted graph too because if $p[i,j]=1$, $p[i,k]=1$ and $p[k,j]=1$ then according to the algorithm $p[i,j]=\min(p[i,j],p[i,k]+p[k,j])=\min(1,2)=1$

And all the other cases are also satisfied. (like if $p[i,j]$ was 0 in last iteration and there exist a path via k)

👍 19 votes -- Saurav Kumar Gupta (1.7k points)

1.6.6 Graph Algorithms: GATE CSE 2003 | Question: 67 top https://gateoverflow.in/954

✔ After applying the shortest path algorithm, check cost of vertex from source to every vertex in G_1. If G_1 is connected all these costs must be 0 as edge weights of subgraph G_1 is 0 and that should be the shortest path. If cost is not 0, to at least one vertex in G_1 (not necessarily G), then G_1 is disconnected.

Answer is B.

👍 62 votes -- Anurag Semwal (6.7k points)

1.6.7 Graph Algorithms: GATE CSE 2003 | Question: 70 top https://gateoverflow.in/957

✔ **D is correct.**

Consider a graph with 2 nodes and one edge from V_1 to V_2,

Running the above algorithm will result in A being

A	1	2
1	1	2
2	1	2

Clearly options B and C are wrong. Since

1. $A[1][1]$ and $A[2][2] > n-1$ and there exists no Hamiltonian cycle. Hence invalid.
2. The longest path between V_1 and V_2 is 1, but $A[1][2]$ is 2, which is invalid. And no path between V_2 and V_1 yet $A[2][1]=1$ // **it should be max cost path between j and k, not path length.**

Hence A or D could be valid.

Now consider a graph with 2 nodes and two edges, one from V_1 and V_2 and other form V_2 and V_1. Running the above algorithm will result in A being

A	1	2
1	2	3
2	3	4

Hence option A is invalid, as $A[i][j]$ can be $> n$.

D is correct

52 votes -- ryan sequeira (3k points)

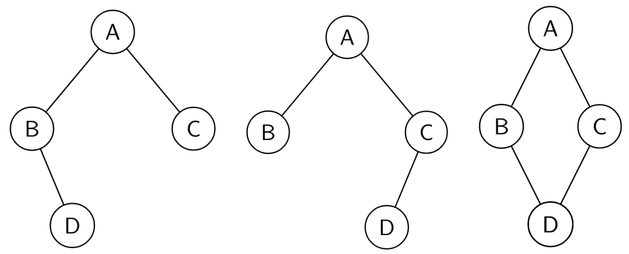
1.6.8 Graph Algorithms: GATE CSE 2004 | Question: 44 top https://gateoverflow.in/1041

✔ Answer is (**B**). In Dijkstra's algorithm at each point we choose the smallest weight edge which starts from any one of the vertices in the shortest path found so far and add it to the shortest path.

27 votes -- gate_asp (615 points)

1.6.9 Graph Algorithms: GATE CSE 2004 | Question: 81 top https://gateoverflow.in/1075

✔ Take a tree for example



G_1 G_2 $G_1 \cup G_2$

A. False. Every vertex of tree (other than leaves) is a cut vertex.
B. True.
C. False. There is no cut-edge (an edge whose removal increases the number of connected components in graph) in $G_1 \cup G_2$.
D. False. $G_1 \cup G_2$, G_1 and $G_2 -$ all three graphs have same the chromatic number of 2.

Now, we have given counter examples for options A,C and D. So, option B is the only possible answer and its proof is given at end.

Correct Answer: Option B

We are given that G_1 and G_2 are connected. So, if we take any two vertices say v_i and v_j there must be path between them in both G_1 and G_2. Now, it is given that $G_1 \cap G_2$ is disconnected. That is, we have at least two vertices v_1 and v_2 such that there is no path between them in $G_1 \cap G_2$.

This means the path between v_1 and v_2 in G_1 and G_2 are **distinct**.

When we have two distinct paths between a pair of vertices in a graph, it forms a cycle.

50 votes -- srestha (85.2k points)

1.6.10 Graph Algorithms: GATE CSE 2005 | Question: 38 top https://gateoverflow.in/1374

✔ Option (D) : Binary heap. $|E|$ decrease key operations and each taking $O\left(\log|V|\right)$ time $+$ $|V|$ extract-min operations each taking $O\left(\log|V|\right)$.

Option (B) : Fibonacci heap. $|E|$ decrease key operations and each taking $O(1)$ time $+$ $|V|$ extract-min operations each taking $O\left(\log|V|\right)$.

Option (A) : Array. Finding min-vertex in each iteration takes $O(V)$ and this needs to be done $|V|$ times.

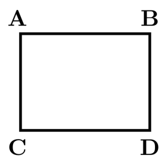
Binomial Heap is same as Binary heap here, as the critical operations are decrease key and extract-min.

Correct Answer: D

64 votes -- Gate Keeda (15.9k points)

1.6.11 Graph Algorithms: GATE CSE 2005 | Question: 82a top https://gateoverflow.in/1404

✔ The answer should be Option **A** because edge e is the lightest safe edge connecting X and Y so the minimum spanning tree of G must contain e (Greedy and optimal choice).
While option (B) might seem correct but it is not always true. One such case is when G is not connected therefore there might not be any path between s and t.
Since the question is about definitely TRUE, (B) is incorrect and (A) is the only correct option



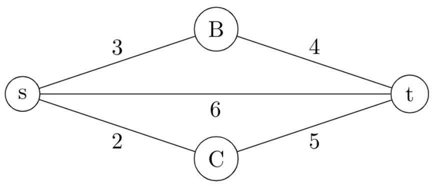
Lets say $AC = 1$, $CD = 2$,$BD = 3$ and $AB = 4$

Then if $s = A$ and $t = B$ then AC is the lightest edge crossing X and Y where $X = A$ and $Y = C, B, D$
But clearly AC is not on the shortest path from A to B. The shortest path is $AB = 4$.

32 votes -- chandan1223 (167 points)

1.6.12 Graph Algorithms: GATE CSE 2005 | Question: 82b top https://gateoverflow.in/82129

✔ Here answer should be **A**.



Here, shortest path will give 6.
Spanning tree contains edges of weights 2,3,4 so congestion in this case is $\max(2, 3, 4)$, that is, 4. For path s to t, overall congestion is $\max(3, 4) = 4$ but total weight is 7.

Option C and D are I think not related to this question.

29 votes -- papesh (18k points)

1.6.13 Graph Algorithms: GATE CSE 2006 | Question: 12 top https://gateoverflow.in/891

✔ Answer is **A:** Queue

We can find single source shortest path in unweighted graph by using Breadth First Search (BFS) algorithm by using "Queue" data structure , in time $O(m + n)$ (i.e. linear with respect to the number of vertices and edges.)

63 votes -- Mithlesh Upadhyay (4.3k points)

1.6.14 Graph Algorithms: GATE CSE 2006 | Question: 48 top https://gateoverflow.in/1824

✔ Let T be a depth-first search tree in an undirected graph G. Vertices u and v are leaves of this tree T. The degrees of both u and v in G are at least 2.

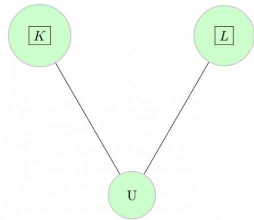
Let's interpret question correctly and draw inferences

> ***Vertices u and v are leaves of this tree T.***

Means (1) **(u,v) is not an edge in the graph** otherwise one would be have been descendent of another and both of them must not be leaves.

(2)If vertices u and v are leaves of tree T, then when DFS was exploring them, after exploring say vertex u, it was unable to find any new unvisited neighbour of u and hence, it had to backtrack the search.So u became leave of T. Same story goes with vertex v.

(3)If degrees of vertices u and v are at least 2 then I say consider the scenario for vertex u



Consider some intermediate vertices K and L which are neighbours of vertex U, and there may be more vertices than K and L, ahead of them connected to either one of them(to K or L) but for simplicity, I consider only two(K and L).

Now, say my DFS algorithm explored vertex K, coloured it, Grey, then it went to vertex U and found it WHITE (means unvisited) so it marks it Grey and makes the edge (K,U) a tree edge. Now DFS examines neighbour of U and finds neighbour L.

Now if the neighbour L was WHITE (Means not visited), DFS would have visited this vertex L and then edge (U,L) would have been marked as tree edge and in this case Vertex, U would no longer be a leave in DFS Tree T.

So, Necessarily my vertex L was visited before U (L Maybe GREY or BLACK in colour) and this would have been connected to vertex K via some other vertices or directly, otherwise, it was impossible for this vertex L to be discovered via path any other than from U to L.

So, what all this means?

Yes Vertex U and it's neighbour are in a cycle and your DFS will mark some edges as back edges.

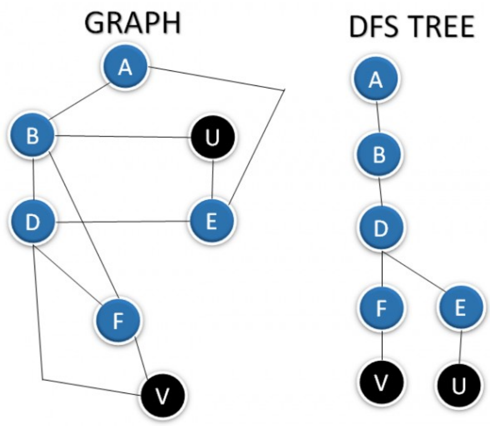
The similar story would hold for vertex V.

So, even if Option had said

"Vertex U and V are involved in cycle with their neighbour" then also this would have been true.

So Option (D) is the answer.

65 votes -- Ayush Upadhyaya (28.3k points)



One diagram, which is eliminating option A, B, C.
Hence **D** is the answer.

59 votes -- Ahwan Mishra (10.2k points)

1.6.15 Graph Algorithms: GATE CSE 2007 | Question: 41 top

https://gateoverflow.in/1239

✔ Dijkastra and Warshall 's algorithm used only for weighted graph.

Both DFS and BFS can be used for finding path between 2 vertices in undirected and unweighted graph but BFS can only give the shortest path as concerned in given question. So, **BFS is answer**.

Note : Finding only path(DFS) and finding shortest path (BFS) matters a lot.

Must Read:

https://www.quora.com/What-are-the-advantages-of-using-BFS-over-DFS-or-using-DFS-over-BFS-What-are-the-applications-and-downsides-of-each

Correct Answer: D

References

👍 36 votes -- Rajesh Pradhan (18.9k points)

1.6.16 Graph Algorithms: GATE CSE 2007 | Question: 5 top

☛ https://gateoverflow.in/31821

✔

> Go with vertex with indegree 0. Remove the vertex with all edges going from it. Repeat the procedure.

We see that 3 cannot come at first because indegree is not 0. **So, D is the answer here.**

ALL other options are in Topological order.

Only 1 and 4 order matter for this question.

👍 26 votes -- Prashant Singh (47.1k points)

1.6.17 Graph Algorithms: GATE CSE 2008 | Question: 45 top

☛ https://gateoverflow.in/457

✔ D. all the vertices. Just simulate the Dijkstra's algorithm on it. Dijkstra's algorithm is not meant for graphs with negative-edge-weight-cycle, but here it does give the correct shortest path.

👍 39 votes -- Arjun Suresh (330k points)

1.6.18 Graph Algorithms: GATE CSE 2008 | Question: 7 top

☛ https://gateoverflow.in/405

✔ Run DFS to find connected components. Its time complexity is $\Theta(m+n)$, hence **(C)** is the answer.

👍 28 votes -- Happy Mittal (8.2k points)

1.6.19 Graph Algorithms: GATE CSE 2009 | Question: 13 top

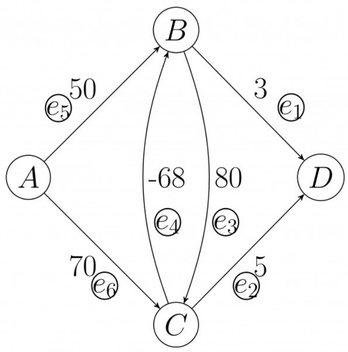
☛ https://gateoverflow.in/1305

✔ Bellman-ford Algorithm

Single source shortest Path $O(VE)$

Relax every edge once in each iteration

$$E \times \underbrace{(V-1)}_{\text{at max } (V-1) \text{ edges can be there}} = E.V - E = O(V.E)$$



	A	B	C	D
	0 null	∞ null	∞ null	∞ null
i=0	0 X	50 A	70 A	∞ X
i=1	0 X	2 C	70 A	53 B
i=2	0 X	2 C	70 A	5 B
i=3	0 X	2 C	70 A	5 B

As we can see that the last step is the verification step. In that step, values remained unchanged. If there was a negative edge weight cycle reachable from source, then at verification step also, those values will be different from the values above.

In case the cycle is not reachable from source then we can see that they will be at ∞ distance(or cost) from the source from the beginning till the last step. As take anything away from the ∞ it will still be infinite.

But it can also be the case that there are some points which are not forming a cycle and are still unreachable from source, those also will be at ∞ distance from the source from the beginning till end.

Hence, we won't be able to make a distinction among the cycle and such vertices. Thus, we say that this algorithm can detect negative edge weight cycles only if they are reachable from the source.

Answer is **option B**

39 votes -- Amar Vashishth (25.2k points)

1.6.20 Graph Algorithms: GATE CSE 2012 | Question: 40 top https://gateoverflow.in/1765

✔ Relaxation at every vertex is as follows:

Note that the next picked vertex corresponds to the next row in Table

	A	B	C	D	E	F	G	T
S	4 $S\to A$	$\boxed{3}$ $\boxed{\mathbf{S\to B}}$	∞	7 $S\to D$	∞	∞	∞	∞
B	$\boxed{4}$ $\boxed{\mathbf{S\to A}}$		∞	7 $S\to D$	∞	∞	∞	∞
A			$\boxed{5}$ $\boxed{\mathbf{S\to A\to C}}$	7 $S\to D$	∞	∞	∞	∞
C				7 $S\to D$	$\boxed{6}$ $\boxed{\mathbf{S\to A\to C\to E}}$	∞	∞	∞
E				$\boxed{7}$ $\boxed{\mathbf{S\to D}}$		∞	8 $S\to A\to C$ $\to E\to G$	10 $S\to A\to C$ $\to E\to T$
D						12 $S\to D\to F$	$\boxed{8}$ $\boxed{\mathbf{S\to A\to C\to E\to G}}$	10 $S\to A\to C$ $\to E\to T$
G						12 $S\to D\to F$		$\boxed{10}$ $\boxed{\mathbf{S\to A\to C\to E\to T}}$
T						$\boxed{12}$ $\boxed{\mathbf{S\to D\to F}}$		

For S to T shortest path is $\boxed{S\to A\to C\to E\to T}$

Option : D

47 votes -- Kalpish Singhal (1.6k points)

1.6.21 Graph Algorithms: GATE CSE 2013 | Question: 19 top https://gateoverflow.in/1441

✔ Time complexity of Bellman-Ford algorithm is $\Theta(|V||E|)$ where $|V|$ is number of vertices and $|E|$ is number of edges. If the graph is complete, the value of $|E|$ becomes $\Theta\left(|V|^2\right)$. So overall time complexity becomes $\Theta\left(|V|^3\right)$. And given here is n vertices. So, the answer ends up to be $\Theta\left(n^3\right)$.

Correct Answer: C

54 votes -- Gate Keeda (15.9k points)

1.6.22 Graph Algorithms: GATE CSE 2014 Set 1 | Question: 13 top https://gateoverflow.in/1779

✔ C. Both PSRQ and SPRQ are topological orderings

i. Apply DFS by choosing P or S as starting vertices
ii. As the vertex gets a finishing time assign it to the head of a linked list
iii. The linked list is your required topological ordering

28 votes -- Akshay Jindal (307 points)

1.6.23 Graph Algorithms: GATE CSE 2016 Set 1 | Question: 11 top https://gateoverflow.in/39669

✔ Here, start with a and end with f.

$$a \_\_\_\_ f$$

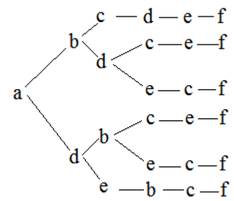
Blank spaces are to be filled with b, c, d, e such that b comes before c, and d comes before e.
Number of ways to arrange b, c, d, e such that b comes before c and d comes before e, will be = $4!/(2!*2!) = 6$

94 votes -- Abhilash Panicker (7.6k points)

In topological sorting all nodes are like tasks and edges show the dependency among the tasks.

Node i to j an edge is there means task i must complete before task j.(in the mean time some other task may get complete after task i and before task j..but task i and j sequence need to be maintained)

Here in following 6 ways all the 6 tasks can get completed.



78 votes -- Rajesh Pradhan (18.9k points)

1.6.24 Graph Algorithms: GATE CSE 2016 Set 2 | Question: 41 top https://gateoverflow.in/39620

✔ We can take extra array of size V^2 as memory is not a constraint.

Do the BFS traversal and for each edge $u - v$ in advance list in graph store the v node address in $a[i][j]$.

For e.g., if we find $1 \rightarrow 2$ as an edge, then store node 2 node address in location $a[i][j]$.

Now once we have stored all the edge $(u - v)$ addresses, then do BFS again.

Now when we encounter $1 \rightarrow 2$, then goto $a[2][1]$ which is having twin address and set this twin pointer for node 2 in list 1.

Do for all edges similarly.

Remember, we have not initialized memory as we will use only slots we need. If we initialize memory it can take $O(V^2)$ time. We can also use one hash table per node to store the node address while doing first traversal.The key to has table for Node u is the vertex v (for edge $u - v$ means in u table it has edge to v and we are storing v address) and value will be v address.

Correct Answer: B

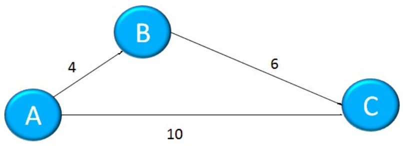
44 votes -- rahul sharma (18.9k points)

1.6.25 Graph Algorithms: GATE CSE 2017 Set 1 | Question: 26 top https://gateoverflow.in/118306

✔ Answer is **A.**

- MST is not unique only when edges are not distinct. Here the edges are distinct. Be careful for the keyword DISTINCT.

- Shortest Path can be different even if the edges are distinct. Example is shown below. Shortest path from A to C is not unique here.



👍 54 votes -- Ahwan Mishra (10.2k points)

1.6.26 Graph Algorithms: GATE CSE 2018 | Question: 43 top

☛ https://gateoverflow.in/204117

✔ **Answer: 109**

Explanation:

We have to find 2 things here, the degree of every vertex(which will be same for all vertices) and number of connected components.

Instead of 100, let's solve this by taking lesser value, say 4.

With 4! vertices, each vertex is a permutation of $\{1, 2, 3, 4\}$. So, we have vertices like $\{1, 2, 3, 4\}$, $\{1, 3, 2, 4\}$, $\{4, 1, 3, 2\}$, ... etc.

Here $\{1, 2, 3, 4\}$will be connected with

$\{2, 1, 3, 4\}$

$\{1, 3, 2, 4\}$

$\{1, 2, 4, 3\}$

To get this list, just take 2 adjacent numbers and swap them. eg. $\{1, 2, 3, 4\}$ swap 1 and 2 to get $\{2, 1, 3, 4\}$.

The given 3 are the only permutations we can get by swapping only 2 adjacent numbers from $\{1, 2, 3, 4\}$. So, the degree of vertex $\{1, 2, 3, 4\}$ will be 3. Similarly for any vertex it's degree will be 3.

Here we got "3" because we can chose any 3 pairs of adjacent numbers. So, with n, we have $n - 1$ adjacent pairs to swap. So, degree will be n-1.

In our question, degree will be $100 - 1 = \mathbf{99}$

Now let's see how many connected components we have.

It will be 1. *Why?*

If one can reach from one vertex to ***any*** other vertex, then that means that the graph is connected.

Now if we start with a vertex say $\{1, 2, 3, 4\}$ we can reach to other vertex, say $\{4, 3, 2, 1\}$ by the following path:

$\{1234\}$ -> $\{1243\}$ -> $\{1423\}$ -> $\{4123\}$ -> $\{4132\}$ -> $\{4312\}$ -> $\{4321\}$

Just take two adjacent numbers and swap them. With this operation you can create any permutation, from any given initial permutation.

This way you can show that from any given vertex we can reach any other vertex. This shows that the graph is connected and the number of connected components is **1**.

y = 99 and z = 1

y + 10z = 99 + 10*1 = 109

👍 162 votes -- Rishabh Gupta (12.5k points)

1.6.27 Graph Algorithms: GATE CSE 2020 | Question: 40 top

☛ https://gateoverflow.in/333191

✔ Correct Answer: A

For any mapping of vertices to real values, the shortest paths won't change. All intermediate nodes values get canceled on any path you take and what you're left with is only the source and destination node values which would add up to cost on any path. Hence, the shortest paths would still be the same.

PS: Option D is wrong because of the "if and only if" clause in it. If it were "if", it would be correct. The condition given is sufficient but not necessary. Hence, "only if" is incorrect in the option. Basically it is saying $f(u)$ would be 0 for all vertices since they're connected to a new vertex s with zero weighted edge. Similarly options B and C are also wrong for the same reason.

👍 19 votes -- Rituraj Joshi (221 points)

1.6.28 Graph Algorithms: GATE CSE 2020 | Question: 49 top https://gateoverflow.in/333182

✔ Vertices are given from 1 to 100 and edge weight between the vertices is absolute values of the difference between the suffix values of the vertices.

So, if we choose the vertices consecutively as $1-2-3-4-5-\ldots-99-100$ we get the spanning tree with minimum weight.

Spanning tree will contain 99 edges since there are 100 vertices and cost of each edge is '1'.

Therefore, weight of spanning tree would be $99*1=99$.

7 votes -- Srinivas_Reddy_Kotla (735 points)

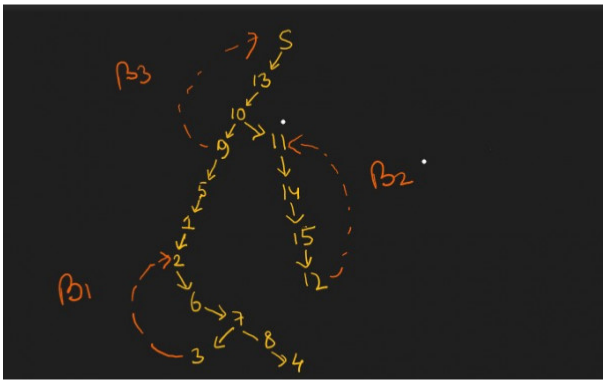
1.6.29 Graph Algorithms: GATE CSE 2021 Set 2 | Question: 46 top https://gateoverflow.in/357494

✔ Lets put numberings to the nodes as follows:

The back edges are those edges from v to u in set (u,v) where u came first (i.e. already explored) in the DFS tree/forest.

- 3 to 2
- 12 to 11
- 9 to S

The DFS tree:



The graph has two cycles one at bottom left and one at bottom right. And every cycle forms **SCC** (Strongly connected component, since from every vertex u, there is a path to v, $\mathbf{u} \rightarrow \mathbf{v}$). C is false

The entire graph doesn't have SCC. So, definitely D is false.

Due to the presence of cycles Topological Sort is not possible.

Hence, the answer is A and B.

D would have been correct if it was **For some pair of vertices u and v there is a directed path from u to v.**

3 votes -- Rishabh Chakraborty (87 points)

1.6.30 Graph Algorithms: GATE CSE 2021 Set 2 | Question: 55 top https://gateoverflow.in/357482

✔ Let $Q(v)$ denote the quality-score of vertex v.

1. $Q(s) = 1$
2. $Q(c) = 1 * 1 \ (s \xrightarrow{1} c)$
3. $Q(f) = 1 * 1 * 9 = 9 \ (s \xrightarrow{1} c \xrightarrow{9} f)$
4. $Q(a) = 1 * 9 = 9 \ (s \xrightarrow{9} a)$
5. $Q(d) = 1 * 9 * 1 = 9 \ (s \xrightarrow{9} a \xrightarrow{1} d)$
6. $Q(g) = 1 * 9 * 1 * 9 = 81 \ (s \xrightarrow{9} a \xrightarrow{1} d \xrightarrow{9} g)$
7. $Q(b) = 1 * 9 * 1 = 9 \ (s \xrightarrow{9} a \xrightarrow{1} b)$
8. $Q(e) = 1 * 9 * 1 * 9 = 81 \ (s \xrightarrow{9} a \xrightarrow{1} d \xrightarrow{9} e)$
9. $Q(t) = 1 * 9 * 1 * 9 * 9 = 729 \ (s \xrightarrow{9} a \xrightarrow{1} d \xrightarrow{9} e \xrightarrow{9} t)$

Therefore, the sum of the quality-scores of all vertices on the graph $= 2 \times 1 + 4 \times 9 + 2 \times 81 + 729 = 929.$

2 votes -- gatecse (62.6k points)

1.6.31 Graph Algorithms: GATE IT 2004 | Question: 56 top

https://gateoverflow.in/3699

✔ Answer is **D.**

A and B produce disconnected components with the GIVEN order in options which is NEVER allowed by prims's algorithm.

C produces connected component every instant a new edge is added BUT when first vertex is chosen(first vertex is chosen randomly) first edge must be the minimum weight edge that is chosen . Therefore, (A, D) MUST be chosen BEFORE (A, B). Therefore, C is FALSE.

28 votes -- Sandeep_Uniyal (6.5k points)

1.6.32 Graph Algorithms: GATE IT 2005 | Question: 15 top

https://gateoverflow.in/3760

✔

1. Bellman-Ford algorithm $\implies$ option(C), $O(nm)$. Assuming n as edges , m as vertices, for every vertex we relax all edges. $m * n$, $O(mn)$.
2. Kruskal's algorithm $\implies$ Remaining Option (A) : $O(m \log n)$.
3. Floyd-Warshall algorithm $\implies$ option (B), Dynamic Programming Algo, $O(N^3)$.
4. Topological sorting $\implies$ option(D), boils down to DFS, $O(V + E)$.

Answer (A).

27 votes -- Akash Kanase (36k points)

1.6.33 Graph Algorithms: GATE IT 2005 | Question: 84a top

https://gateoverflow.in/3856

✔ If there is a sink in the graph, the adjacency matrix will contain all 1's (except diagonal) in one column and all 0's (except diagonal) in the corresponding row of that vertex. The given algorithm is a smart way of doing this as it finds the sink in $O(n)$ time complexity.

The first part of the code, is finding if there is any vertex which doesn't have any outgoing edge to any vertex coming after it in adjacency matrix. The smart part of the code is E_2, which makes rows skip when there is no edge from i to it, making it impossible for them to form a sink. This is done through

- E_1 :$!A[i][j]$ and $E_2 : i = j$;

E_1 makes sure that there is no edge from i to j and i is a potential sink till $A[i][j]$ becomes 1. If $A[i][j]$ becomes 1, i can no longer be a sink, similarly all previous j can also not be a sink (as there was no edge from i to them and a sink requires an edge from all other vertices). Now, the next potential candidate for sink is j. So, in E_2, we must make $i = j$.

So, **answer is (C)**

For E_3, https://gateoverflow.in/3857/gate2005-it_84b

References

45 votes -- Arjun Suresh (330k points)

1.6.34 Graph Algorithms: GATE IT 2005 | Question: 84b top https://gateoverflow.in/3857

✔ If there is a sink in the graph, the adjacency matrix will contain all 1s (except diagonal) in one column and all 0s (except diagonal) in the corresponding row of that vertex. The given algorithm is a smart way of doing this as it finds the sink in $O(n)$ time complexity.

The first part of the code, is finding if there is any vertex which does not have any outgoing edge to any vertex coming after it in adjacency matrix. The smart part of the code is E_2, which makes rows skip when there is no edge from i to it, making it impossible them to form a sink. This is done through

E_1 : $!A[i][j]$
and
$E_2 : i = j$;

E_1 makes sure that there is no edge from i to j and i is a potential sink till $A[i][j]$ becomes 1. If $A[i][j]$ becomes 1, i can no longer be a sink, similarly all previous j can also not be a sink (as there was no edge from i to them and a sink requires an edge from all other vertices). Now, the next potential candidate for sink is j. So, in E_2, we must make $i = j$.

Now, the loop breaks when we found a potential sink- that is a vertex which does not have any outgoing edge to any coming after it in adjacency matrix. So, if the column in which this vertex comes is all 1s and the row is all 0s (except diagonal), this is the sink. Otherwise there is no sink in the graph. So, E_3 is checking this condition.

But in the code flag is used for storing the state that sink is present or not. And as per the usage of flag in code, by default sink is considered present. So, the condition in E_3 must make flag = 0, if the found i is not a sink. So, the condition should be:

$A[i][j] \ || \ !A[j][i]$

So, **(D) is the answer**.

48 votes -- Arjun Suresh (330k points)

1.6.35 Graph Algorithms: GATE IT 2006 | Question: 46 top https://gateoverflow.in/3589

✔ Here the answer is **B**.

A graph is said to be **strongly connected** if every vertex is reachable from every other vertex.

The strongly connected component is always maximal that is if x is strongly connected component there should not exist another strongly connected component which contains x.

If we take R as a strongly connected component but which is part of $PQRS$ and $PQRS$ is part of $PQRSVT$.

39 votes -- papesh (18k points)

1.6.36 Graph Algorithms: GATE IT 2007 | Question: 3, UGCNET-June2012-III: 34 top https://gateoverflow.in/3434

✔ C. Weight $(u, v) \geq 12$

If weight $(u, v) < 12$, then the min. weight of (s, v) =weight of $(s, u)+$ weight of $(u, v) = 53 + (< 12)$ will be less than 65.

36 votes -- Arjun Suresh (330k points)

1.7 Graph Search (15) top

1.7.1 Graph Search: GATE CSE 2000 | Question: 1.13 top https://gateoverflow.in/636

The most appropriate matching for the following pairs

X: depth first search	1: heap
Y: breadth first search	2: queue
Z: sorting	3: stack

is:

A. X - 1, Y - 2, Z - 3
B. X - 3, Y - 1, Z - 2

C. X - 3, Y - 2, Z - 1
D. X - 2, Y - 3, Z - 1

gate2000-cse algorithms easy graph-algorithms graph-search

Answer

1.7.2 Graph Search: GATE CSE 2001 | Question: 2.14 top

☛ https://gateoverflow.in/732

Consider an undirected, unweighted graph G. Let a breadth-first traversal of G be done starting from a node r. Let $d(r,u)$ and $d(r,v)$ be the lengths of the shortest paths from r to u and v respectively in G. If u is visited before v during the breadth-first traversal, which of the following statements is correct?

A. $d(r,u) < d(r,v)$
B. $d(r,u) > d(r,v)$
C. $d(r,u) \leq d(r,v)$
D. None of the above

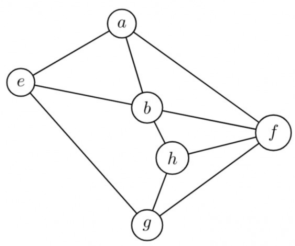
gate2001-cse algorithms graph-algorithms normal graph-search

Answer

1.7.3 Graph Search: GATE CSE 2003 | Question: 21 top

☛ https://gateoverflow.in/911

Consider the following graph:



Among the following sequences:

I. abeghf
II. abfehg
III. abfhge
IV. afghbe

Which are the depth-first traversals of the above graph?

A. I, II and IV only
B. I and IV only
C. II, III and IV only
D. I, III and IV only

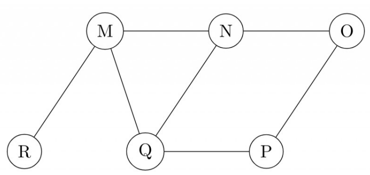
gate2003-cse algorithms graph-algorithms normal graph-search

Answer

1.7.4 Graph Search: GATE CSE 2008 | Question: 19 top

☛ https://gateoverflow.in/417

The Breadth First Search algorithm has been implemented using the queue data structure. One possible order of visiting the nodes of the following graph is:



A. MNOPQR
B. NQMPOR
C. QMNPRO

D. QMNPOR

gate2008-cse normal algorithms graph-algorithms graph-search

Answer

1.7.5 Graph Search: GATE CSE 2014 Set 1 | Question: 11 top

https://gateoverflow.in/1771

Let G be a graph with n vertices and m edges. What is the tightest upper bound on the running time of Depth First Search on G, when G is represented as an adjacency matrix?

A. $\Theta(n)$
B. $\Theta(n+m)$
C. $\Theta(n^2)$
D. $\Theta(m^2)$

gate2014-cse-set1 algorithms graph-algorithms normal graph-search

Answer

1.7.6 Graph Search: GATE CSE 2014 Set 2 | Question: 14 top

https://gateoverflow.in/1969

Consider the tree arcs of a BFS traversal from a source node W in an unweighted, connected, undirected graph. The tree T formed by the tree arcs is a data structure for computing

A. the shortest path between every pair of vertices.
B. the shortest path from W to every vertex in the graph.
C. the shortest paths from W to only those nodes that are leaves of T.
D. the longest path in the graph.

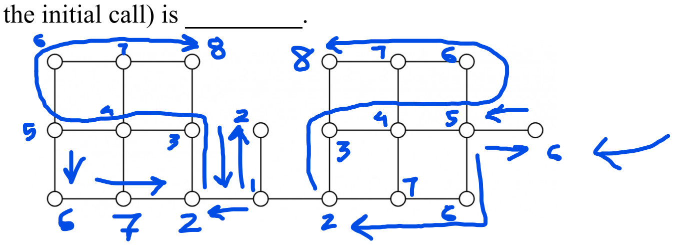
gate2014-cse-set2 algorithms graph-algorithms normal graph-search

Answer

1.7.7 Graph Search: GATE CSE 2014 Set 3 | Question: 13 top

https://gateoverflow.in/2047

Suppose depth first search is executed on the graph below starting at some unknown vertex. Assume that a recursive call to visit a vertex is made only after first checking that the vertex has not been visited earlier. Then the maximum possible recursion depth (including the initial call) is __________.



min depth = 8

gate2014-cse-set3 algorithms graph-algorithms numerical-answers normal graph-search

Answer

1.7.8 Graph Search: GATE CSE 2015 Set 1 | Question: 45 top

https://gateoverflow.in/8321

Let $G=(V,E)$ be a simple undirected graph, and s be a particular vertex in it called the source. For $x \in V$, let $d(x)$ denote the shortest distance in G from s to x. A breadth first search (BFS) is performed starting at s. Let T be the resultant BFS tree. If (u,v) is an edge of G that is not in T, then which one of the following CANNOT be the value of $d(u)-d(v)$?

A. -1
B. 0
C. 1
D. 2

gate2015-cse-set1 algorithms graph-algorithms normal graph-search

Answer

1.7.9 Graph Search: GATE CSE 2016 Set 2 | Question: 11 top https://gateoverflow.in/39563

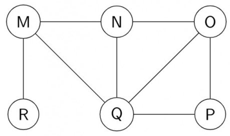
Breadth First Search (BFS) is started on a binary tree beginning from the root vertex. There is a vertex t at a distance four from the root. If t is the n^{th} vertex in this BFS traversal, then the maximum possible value of n is ________

gate2016-cse-set2 algorithms graph-algorithms normal numerical-answers graph-search

Answer

1.7.10 Graph Search: GATE CSE 2017 Set 2 | Question: 15 top https://gateoverflow.in/118196

The Breadth First Search (BFS) algorithm has been implemented using the queue data structure. Which one of the following is a possible order of visiting the nodes in the graph below?



A. MNOPQR
B. NQMPOR
C. QMNROP
D. POQNMR

gate2017-cse-set2 algorithms graph-algorithms graph-search

Answer

1.7.11 Graph Search: GATE CSE 2018 | Question: 30 top https://gateoverflow.in/204104

Let G be a simple undirected graph. Let T_D be a depth first search tree of G. Let T_B be a breadth first search tree of G. Consider the following statements.

I. No edge of G is a cross edge with respect to T_D. (A cross edge in G is between two nodes neither of which is an ancestor of the other in T_D).
II. For every edge (u, v) of G, if u is at depth i and v is at depth j in T_B, then $|i - j| = 1$.

Which of the statements above must necessarily be true?

A. I only
B. II only
C. Both I and II
D. Neither I nor II

gate2018-cse algorithms graph-algorithms graph-search normal

Answer

1.7.12 Graph Search: GATE IT 2005 | Question: 14 top https://gateoverflow.in/3759

In a depth-first traversal of a graph G with n vertices, k edges are marked as tree edges. The number of connected components in G is

A. k
B. $k+1$
C. $n-k-1$
D. $n-k$

gate2005-it algorithms graph-algorithms normal graph-search

Answer

1.7.13 Graph Search: GATE IT 2006 | Question: 47 top https://gateoverflow.in/3590

Consider the depth-first-search of an undirected graph with 3 vertices P, Q, and R. Let discovery time $d(u)$ represent the time instant when the vertex u is first visited, and finish time $f(u)$ represent the time instant when the vertex u is last visited. Given that

$d(P) = 5$ units	$f(P) = 12$ units
$d(Q) = 6$ units	$f(Q) = 10$ units
$d(R) = 14$ unit	$f(R) = 18$ units

Which one of the following statements is TRUE about the graph?

A. There is only one connected component
B. There are two connected components, and P and R are connected
C. There are two connected components, and Q and R are connected
D. There are two connected components, and P and Q are connected

gate2006-it algorithms graph-algorithms normal graph-search

Answer

1.7.14 Graph Search: GATE IT 2007 | Question: 24 top

https://gateoverflow.in/3457

A depth-first search is performed on a directed acyclic graph. Let $d[u]$ denote the time at which vertex u is visited for the first time and $f[u]$ the time at which the DFS call to the vertex u terminates. Which of the following statements is always TRUE for all edges (u, v) in the graph ?

A. $d[u] < d[v]$
B. $d[u] < f[v]$
C. $f[u] < f[v]$
D. $f[u] > f[v]$

gate2007-it algorithms graph-algorithms normal graph-search

Answer

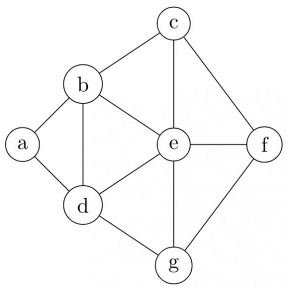
1.7.15 Graph Search: GATE IT 2008 | Question: 47 top

https://gateoverflow.in/3357

Consider the following sequence of nodes for the undirected graph given below:

1. $a\ b\ e\ f\ d\ g\ c$
2. $a\ b\ e\ f\ c\ g\ d$
3. $a\ d\ g\ e\ b\ c\ f$
4. $a\ d\ b\ c\ g\ e\ f$

A Depth First Search (DFS) is started at node a. The nodes are listed in the order they are first visited. Which of the above is/are possible output(s)?



A. 1 and 3 only
B. 2 and 3 only
C. $2, 3$ and 4 only
D. $1, 2$ and 3 only

gate2008-it algorithms graph-algorithms normal graph-search

Answer

Answers: Graph Search

1.7.1 Graph Search: GATE CSE 2000 | Question: 1.13 top https://gateoverflow.in/636

✔ Answer is C.

X - 3 **DFS** uses **stack** implicitly
Y - 2 **BFS** uses **queue** explicitly in Algo
Z - 1 **Heap-Heapsort**

20 votes -- Akash Kanase (36k points)

1.7.2 Graph Search: GATE CSE 2001 | Question: 2.14 top https://gateoverflow.in/732

✔ Answer is (C).

BFS is used to count shortest path from source (If all path costs are 1)

Now, if u is visited before v it means 2 things:

1. Either u is closer to v, or,
2. If u & v are same distance from r, then our BFS algo chose to visit u before v.

43 votes -- Akash Kanase (36k points)

1.7.3 Graph Search: GATE CSE 2003 | Question: 21 top https://gateoverflow.in/911

✔ For GATE purpose, without actually applying DFS, we can answer by just seeing options.
In DFS, we go in depth first i.e., one node to another in depth first order.

Here, $abfehg$ is not possible as we can not go from f to e directly.
Thus, option (D) is correct.

In all the other options we can reach directly from the node to the next node.

So, just visualize and do.

22 votes -- Monanshi Jain (7k points)

1.7.4 Graph Search: GATE CSE 2008 | Question: 19 top https://gateoverflow.in/417

✔

A. MNOPQR : If you try to run BFS, after M, you must traverse NQR (In some order). Here, P is traversed before Q, which is wrong.
B. NQMPOR: This is also not BFS. P is traversed before O.
C. QMNPRO: Correct.
D. QMNPOR: Incorrect. Because R needs to be traversed before O. (Because M is ahead of N in queue).

Answer: C

28 votes -- Akash Kanase (36k points)

1.7.5 Graph Search: GATE CSE 2014 Set 1 | Question: 11 top https://gateoverflow.in/1771

✔ [Depth First Search of a graph](#) takes $O(m+n)$ time when the graph is represented using adjacency list. In adjacency matrix representation, graph is represented as an $n*n$ matrix. To do DFS, for every vertex, we traverse the row corresponding to that vertex to find all adjacent vertices (In adjacency list representation we traverse only the adjacent vertices of the vertex). Therefore time complexity becomes $O(n^2)$.

Correct Answer: C

References

24 votes -- Divya Bharti (8.8k points)

1.7.6 Graph Search: GATE CSE 2014 Set 2 | Question: 14 top

https://gateoverflow.in/1969

✔ BFS always has a starting node. It does not calculate shortest path between every pair but it computes shortest path between W and any other vertex.

Correct Answer: B

57 votes -- Digvijay (44.9k points)

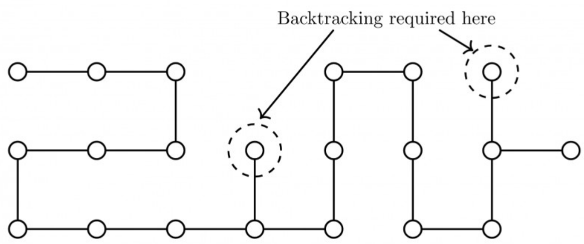
1.7.7 Graph Search: GATE CSE 2014 Set 3 | Question: 13 top

https://gateoverflow.in/2047

✔ Total 21 nodes are there. 2 nodes require back track here in this question.

So, max recursion depth is $21 - 2 = 19$

(Do DFS from extreme ends such that max recursion depth will occur. i.e., take leftmost top node as initial node for DFS as shown in below image)



Note:- Backtrack means it reduces recursion depth in stack.

57 votes -- Rajesh Pradhan (18.9k points)

1.7.8 Graph Search: GATE CSE 2015 Set 1 | Question: 45 top

https://gateoverflow.in/8321

✔ 2 is the answer.

$d(u) - d(v) = 0$ is possible when both u and v have an edge from from a common node t and t is in the shortest path from s to u and v.

$d(u) - d(v) = 1$ is possible when v and another node t are in the shortest path from s to u and both t and v are siblings- same distance from s to both t and v causing $t - u$ edge to be in BFS tree and not $v - u$.

$d(u) - d(v) = -1$ is possible as explained above by interchanging u and v.

$d(u) - d(v) = 2$ is not possible. This is because on BFS traversal we either visit u first or v. Let's take u first. Now, we put all neighbors of u on queue. Since v is a neighbour and v is not visited before as assumed, $d(v)$ will become $d(u) + 1$. Similarly, for v being visited first.

51 votes -- Arjun Suresh (330k points)

1.7.9 Graph Search: GATE CSE 2016 Set 2 | Question: 11 top

https://gateoverflow.in/39563

✔ No of nodes at level 0(root) of tree $\Rightarrow 1$

No of nodes at level 1 of tree $\Rightarrow 2$

No of nodes at level 2 of tree $\Rightarrow 4$

No of nodes at level 3 of tree $\Rightarrow 8$

No of nodes at level 4 of tree $\Rightarrow 16$

Last node in level 4th is the node we are looking for $\Rightarrow 1 + 2 + 4 + 8 + 16 \Rightarrow 31$

43 votes -- Akash Kanase (36k points)

1.7.10 Graph Search: GATE CSE 2017 Set 2 | Question: 15 top

https://gateoverflow.in/118196

✔

> In BFS, starting from a node, we traverse all node adjacent to it at first then repeat same for next nodes.

Here, we can see that only option (D) is following BFS sequence properly.

- As per BFS, if we start from M then RQN (immediate neighbors of M) have to come after it in any order but in A here, O comes in between. So, it is not BFS.
- As per BFS, if we start from N then QMO has to come after it in any order but in B here, P comes. So, it is not BFS.
- As per BFS, if we start from Q then $MNOP$ has to come after it in any order but in C here, R comes. So, it is not BFS.

But D is following the sequences.

So, **D** is the correct answer.

19 votes -- Aboveallplayer (12.5k points)

1.7.11 Graph Search: GATE CSE 2018 | Question: 30 top

https://gateoverflow.in/204104

✔

I. **Undirected graph** cant have cross edges in DFS forest. (Only directed graphs can have) (Hence **I is True**) (If you do not agree just try it with taking examples. You cant draw)

II. Just draw a triangle SAB. Source is S. Vertex A and B are at same level hence distance 1. So here $|i - j| = 0$. (Hence **II is false**)

Hence answer is **(A).**

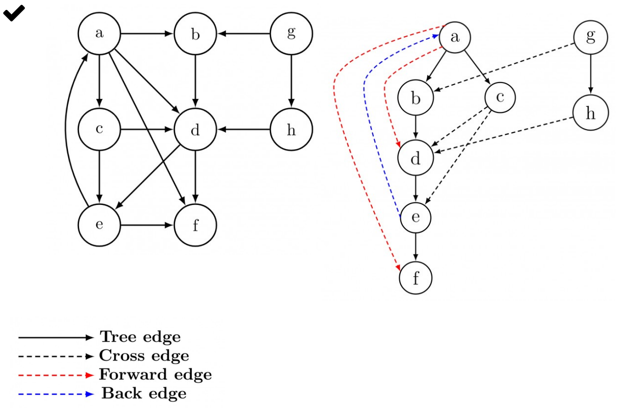
Anyway, **II** is simple to understand from the above explanation.
Those who did not get **I**, you may see Theorem 22.10 in Cormen)

35 votes -- Ahwan Mishra (10.2k points)

1.7.12 Graph Search: GATE IT 2005 | Question: 14 top

https://gateoverflow.in/3759

✔



Tree edges are those edges which appear in the final DFS forest. For example in case of connected graph (i.e. resulting DFS forest containing only one tree), if we run DFS over it, edges belonging to the resulting DFS tree are called tree edges.

Let us assume the graph has x number of connected (or strongly connected in case of a directed graph) components. And assume 1st component has K_1 tree edges, 2nd component has K_2 tree edges and xth component has K_x tree edges.

Such that $K_1 + K_2 + K_3 + \ldots + K_x = K$ (= total)

Or in other way we can imagine like, the final DFS forest has x trees and those trees are having K_1 , K_2 , $K_3, \ldots, K_x$ edges respectively.

Now we know that a tree having K_x edges contains $K_x + 1$ nodes. Similarly a tree having K_1 edges contains $K_1 + 1$ nodes, etc. and so on.

So, summation of nodes in each tree $= n$

$$(K_1 + 1) + (K_2 + 1) + (K_3 + 1) + \ldots + (K_x + 1) = n \quad \implies (K_1 + K_2 + K_3 + \ldots + K_x) + x = n \quad \implies x = n$$

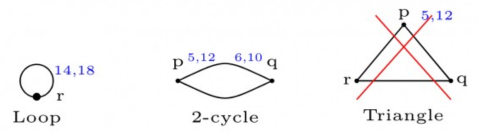
Correct Answer: D

👍 88 votes -- Debashish Deka (40.7k points)

1.7.13 Graph Search: GATE IT 2006 | Question: 47 top

☛ https://gateoverflow.in/3590

✔



As seen in question, after 10 we have to go for p again and since p is finished and then r is started it means r must be disconnected. If there is an edge from q to r then r must be visited before q and p end.

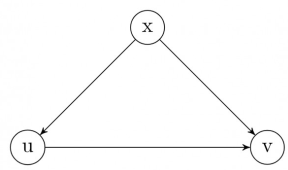
D is answer.

👍 26 votes -- Prashant Singh (47.1k points)

1.7.14 Graph Search: GATE IT 2007 | Question: 24 top

☛ https://gateoverflow.in/3457

✔



I'm gonna disprove all wrong options here:

A. $d[u] < d[v]$, Counter Example $\implies$ Well if we directly start DFS on V first, then I call DFS on X which visits U.
B. $d[u] < f[v]$, Counter example $\implies$ Same as A
C. $f[u] < f[v]$, Counter example $\implies$ Same as A again

So, answer is D.

👍 41 votes -- Akash Kanase (36k points)

1.7.15 Graph Search: GATE IT 2008 | Question: 47 top

☛ https://gateoverflow.in/3357

✔ **Answer: B**

1. After f is visited, c or g should be visited next. So, the traversal is incorrect.
4. After c is visited, e or f should be visited next. So, the traversal is incorrect.

2 and 3 are correct.

👍 20 votes -- Rajarshi Sarkar (27.8k points)

1.8 Greedy Algorithm (7) top

1.8.1 Greedy Algorithm: GATE CSE 1999 | Question: 2.20 top

☛ https://gateoverflow.in/466

The minimum number of record movements required to merge five files A (with 10 records), B (with 20 records), C (with 15 records), D (with 5 records) and E (with 25 records) is:

A. 165
B. 90
C. 75
D. 65

gate1999 algorithms normal greedy-algorithm

Answer

1.8.2 Greedy Algorithm: GATE CSE 2003 | Question: 69 top

☛ https://gateoverflow.in/956

The following are the starting and ending times of activities A, B, C, D, E, F, G and H respectively in chronological order: "a_s b_s c_s a_e d_s c_e e_s f_s b_e d_e g_s e_e f_e h_s g_e h_e" . Here, x_s denotes the starting time and x_e denotes the ending time of activity X. We need to schedule the activities in a set of rooms available to us. An activity can be scheduled in a room only if the room is reserved for the activity for its entire duration. What is the minimum number of rooms required?

A. 3
B. 4
C. 5
D. 6

gate2003-cse algorithms normal greedy-algorithm

Answer

1.8.3 Greedy Algorithm: GATE CSE 2005 | Question: 84a top https://gateoverflow.in/1406

We are given 9 tasks $T_1, T_2, \dots, T_9$. The execution of each task requires one unit of time. We can execute one task at a time. Each task T_i has a profit P_i and a deadline d_i. Profit P_i is earned if the task is completed before the end of the d_i^{th} unit of time.

Task	T_1	T_2	T_3	T_4	T_5	T_6	T_7	T_8	T_9
Profit	15	20	30	18	18	10	23	16	25
Deadline	7	2	5	3	4	5	2	7	3

Are all tasks completed in the schedule that gives maximum profit?

A. All tasks are completed
B. T_1 and T_6 are left out
C. T_1 and T_8 are left out
D. T_4 and T_6 are left out

gate2005-cse algorithms greedy-algorithm process-scheduling normal

Answer

1.8.4 Greedy Algorithm: GATE CSE 2005 | Question: 84b top https://gateoverflow.in/82514

We are given 9 tasks $T_1, T_2, \dots, T_9$. The execution of each task requires one unit of time. We can execute one task at a time. Each task T_i has a profit P_i and a deadline d_i. Profit P_i is earned if the task is completed before the end of the d_i^{th} unit of time.

Task	T_1	T_2	T_3	T_4	T_5	T_6	T_7	T_8	T_9
Profit	15	20	30	18	18	10	23	16	25
Deadline	7	2	5	3	4	5	2	7	3

What is the maximum profit earned?

A. 147
B. 165
C. 167
D. 175

gate2005-cse algorithms greedy-algorithm process-scheduling normal

Answer

1.8.5 Greedy Algorithm: GATE CSE 2007 | Question: 76 top https://gateoverflow.in/1271

Suppose the letters a, b, c, d, e, f have probabilities $\frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{16}, \frac{1}{32}, \frac{1}{32}$, respectively.

Which of the following is the Huffman code for the letter a, b, c, d, e, f?

A. 0, 10, 110, 1110, 11110, 11111
B. 11, 10, 011, 010, 001, 000
C. 11, 10, 01, 001, 0001, 0000
D. 110, 100, 010, 000, 001, 111

gate2007-cse algorithms greedy-algorithm normal

Answer

1.8.6 Greedy Algorithm: GATE CSE 2018 | Question: 48 top

https://gateoverflow.in/204123

Consider the weights and values of items listed below. Note that there is only one unit of each item.

Item number	Weight (in Kgs)	Value (in rupees)
1	10	60
2	7	28
3	4	20
4	2	24

The task is to pick a subset of these items such that their total weight is no more than 11 Kgs and their total value is maximized. Moreover, no item may be split. The total value of items picked by an optimal algorithm is denoted by V_{opt}. A greedy algorithm sorts the items by their value-to-weight ratios in descending order and packs them greedily, starting from the first item in the ordered list. The total value of items picked by the greedy algorithm is denoted by V_{greedy}.

The value of $V_{opt} - V_{greedy}$ is ____

gate2018-cse algorithms greedy-algorithm numerical-answers

Answer

1.8.7 Greedy Algorithm: GATE IT 2006 | Question: 48 top

https://gateoverflow.in/3591

The characters a to h have the set of frequencies based on the first 8 Fibonacci numbers as follows

$a:1, b:1, c:2, d:3, e:5, f:8, g:13, h:21$

A Huffman code is used to represent the characters. What is the sequence of characters corresponding to the following code?

110111100111010

A. $fdheg$
B. $ecgdf$
C. $dchfg$
D. $fehdg$

gate2006-it algorithms greedy-algorithm normal

Answer

Answers: Greedy Algorithm

1.8.1 Greedy Algorithm: GATE CSE 1999 | Question: 2.20 top

https://gateoverflow.in/466

✔ Arrange files in increasing order of records:

$\boxed{5}$	$\boxed{10}$	$\boxed{15}$	$\boxed{20}$	$\boxed{25}$
D	A	C	B	E

75

30 45

15 $\boxed{C}$ 15 $\boxed{B}$ 20 $\boxed{E}$ 25

$\boxed{D}$ 5 $\boxed{A}$ 10

No. of movements $= 15 + 30 + 45 + 75 = 165.$

Correct Answer: A

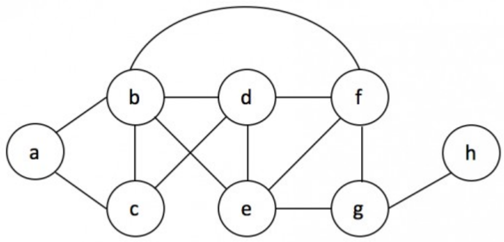
31 votes -- Pooja Palod (24.1k points)

1.8.2 Greedy Algorithm: GATE CSE 2003 | Question: 69 top

https://gateoverflow.in/956

✔ Solution: B

The problem can be modeled as a graph coloring problem. Construct a graph with one node corresponding to each activity A, B, C, D, E, F, G and H. Connect the activities that occur between the start and end time of an activity. Now, the chromatic number of the graph is the number of rooms required.



48 votes -- Gowthaman Arumugam (1.2k points)

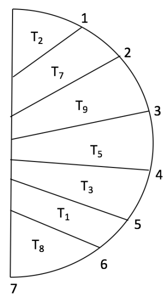
1.8.3 Greedy Algorithm: GATE CSE 2005 | Question: 84a top

https://gateoverflow.in/1406

✔

Task	Profit	Deadline	
T_3	30	5	✓
T_9	25	3	✓
T_7	23	2	✓
T_2	20	2	✓
T_5	18	4	✓
T_4	18	3	✗
T_8	16	7	✓
T_1	15	7	✓
T_6	10	2	✗

Step -1 Sort the tasks in decreasing order of profit and if any conflict arises between two or more tasks,resolve them by sorting them on basis of having greater deadline first(Because we have more time to complete the task with greater deadline and same profit).



Step 2- Since Maximum deadline given is 7, so we consider we have 7 time slots ranging from $0-7$ where a task T_i having deadline say 2 can be filled in slots either $0-1$ or $1-2$ and not beyond 2 because this task has deadline of 2 time units, so this task has to be completed by at most time $T = 2$.

Now according to question, since Each task completes in Unit time, so a single tasks takes only one slot as shown.

Now Take the first task in the list i.e. T_3 which has a deadline of 5, so it can be completed in maximum 5 time units, so place it in slot $4-5$ which is the maximum deadline by which this task can be completed.

Task T_9 with deadline 3 is similarly placed in slot $2-3$.

Task T_7 with deadline 2 is placed in slot $1-2$.

Now for task T_2 having deadline 2 can be placed in either $0-1$ or $1-2$ (Occupied by T_7). So T_2 will occupy slot $0-1$.

Task T_5 with deadline 4 is placed in slot $3-4$.

Now comes task T_4 which has deadline 3 can be put in slots $0-1$ or $1-2$ or $2-3$ and not beyond that.Unfortunately, all such slots are occupied so $\mathbf{T_4}$ **will be left out.**

Task T_8 with deadline 7 goes in slot $6-7$.

Task T_1 with deadline 7 can be placed in slot $5-6$.

Now all time slots are **full.**

So, Task $\mathbf{T_6}$ **will be left out.**

So, option (d) is the answer.

37 votes

-- Ayush Upadhyaya (28.3k points)

1.8.4 Greedy Algorithm: GATE CSE 2005 | Question: 84b top https://gateoverflow.in/82514

The most important statement in question is

> each task requires one unit of time

This shows that we can greedily choose the better task and that should give us the optimal solution. The best task would be the one with maximum profit. Thus we can sort the tasks based on deadline and then profit as follows:

Task	T_7	T_2	T_9	T_4	T_5	T_3	T_6	T_8	T_1
Deadline	2	2	3	3	4	5	5	7	7

$$0 \overset{T_7}{-} 1 \overset{T_2}{-} 2 \overset{T_9}{-} 3 \overset{T_5}{-} 4 \overset{T_3}{-} 5 \overset{T_8}{-} 6 \overset{T_1}{-} 7$$

Thus, T_4 and T_6 are left.

So, maximum profit will not include those of T_4 and T_6 and will be $= 15 + 20 + 30 + 18 + 16 + 23 + 25 = 147$

A is answer

27 votes

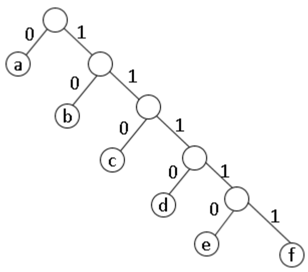
-- Prashant Singh (47.1k points)

1.8.5 Greedy Algorithm: GATE CSE 2007 | Question: 76 top https://gateoverflow.in/1271

Based on the probabilities, we can say the probable frequency of the letters will be

16, 8, 4, 2, 1, 1

Now, the Huffman tree can be constructed as follows:



So, **A** is the answer for 76.

https://en.wikipedia.org/wiki/Huffman_coding

References

32 votes

-- Arjun Suresh (330k points)

1.8.6 Greedy Algorithm: GATE CSE 2018 | Question: 48 top https://gateoverflow.in/204123

V_{opt} is clearly 60. You can go for brute force or by normal intuition you can get it.

Now solving for V_{greedy}.

Item name	Weight (in Kgs)	Value (in Rupees)	Value/Weight
1	10	60	6
2	7	28	4
3	4	20	5
4	2	24	12

Sort them in descending order of Value/Weight as per the question.

Item name	Weight (in Kgs)	Value (in Rupees)	Value/Weight
4	2	24	12
1	10	60	6
3	4	20	5
2	7	28	4

Now start picking items.(Note: You cannot take a fraction of the given weight as per the question). Max weight size is given as 11(Inclusive).

- Item 4 is picked. Weight remaining $= 11 - 2 = 9$kg.
- Item 1 cannot be picked as $10\text{kg} > 9\text{kg}$.
- Item 3 can be picked as $4\text{kg} < 9\text{kg}$. Weight Remaining $= 9 - 4 = 5\text{kg}$
- Item 2 cannot be picked as $7\text{kg} > 5\text{kg}$.

So, item 4 and Item 3 are picked. Their values are 24 and 20 respectively.

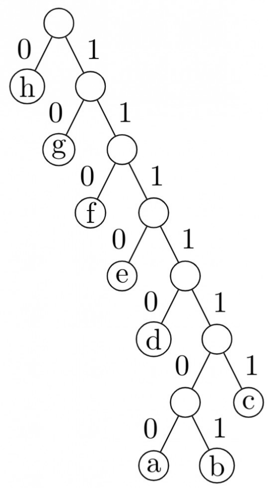
$\implies V_{greedy} = 24 + 20 = 44.$

$V_{optimal} - V_{greedy} = 60 - 44 = 16.$

42 votes -- Ruturaj Mohanty (3.1k points)

1.8.7 Greedy Algorithm: GATE IT 2006 | Question: 48 top https://gateoverflow.in/3591

✔ Answer is **A**. Huffman's tree is as follows. The two least frequent characters are taken as the children of a newly made node and the frequency of the newly made node is made equal to the sum of those two child nodes. Then the same procedure is repeated till all nodes are finished.



$110111100111010 = 110\ 11110\ 0\ 1110\ 10 = fdheg$

43 votes -- Arjun Suresh (330k points)

1.9 Hashing (4) top

1.9.1 Hashing: GATE CSE 1990 | Question: 13b top https://gateoverflow.in/86225

Consider a hash table with chaining scheme for overflow handling:

i. What is the worst-case timing complexity of inserting n elements into such a table?

ii. For what type of instance does this hashing scheme take the worst-case time for insertion?

gate1990 hashing algorithms descriptive

Answer

1.9.2 Hashing: GATE CSE 2020 | Question: 23 top

https://gateoverflow.in/333208

Consider a double hashing scheme in which the primary hash function is $h_1(k) = k \bmod 23$, and the secondary hash function is $h_2(k) = 1 + (k \bmod 19)$. Assume that the table size is 23. Then the address returned by probe 1 in the probe sequence (assume that the probe sequence begins at probe 0) for key value $k = 90$ is_____________.

gate2020-cse numerical-answers algorithms hashing

Answer

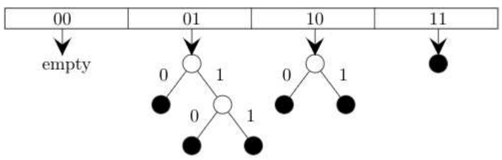
1.9.3 Hashing: GATE CSE 2021 Set 1 | Question: 47 top

https://gateoverflow.in/357404

Consider a *dynamic* hashing approach for 4-bit integer keys:

1. There is a main hash table of size 4.
2. The 2 least significant bits of a key is used to index into the main hash table.
3. Initially, the main hash table entries are empty.
4. Thereafter, when more keys are hashed into it, to resolve collisions, the set of all keys corresponding to a main hash table entry is organized as a binary tree that grows on demand.
5. First, the 3^{rd} least significant bit is used to divide the keys into left and right subtrees.
6. To resolve more collisions, each node of the binary tree is further sub-divided into left and right subtrees based on the 4^{th} least significant bit.
7. A split is done only if it is needed, i.e., only when there is a collision.

Consider the following state of the hash table.



Which of the following sequences of key insertions can cause the above state of the hash table (assume the keys are in decimal notation)?

A. $5, 9, 4, 13, 10, 7$
B. $9, 5, 10, 6, 7, 1$
C. $10, 9, 6, 7, 5, 13$
D. $9, 5, 13, 6, 10, 14$

gate2021-cse-set1 multiple-selects algorithms hashing

Answer

1.9.4 Hashing: GATE IT 2005 | Question: 16 top

https://gateoverflow.in/3761

A hash table contains 10 buckets and uses linear probing to resolve collisions. The key values are integers and the hash function used is key$\%10$. If the values $43, 165, 62, 123, 142$ are inserted in the table, in what location would the key value 142 be inserted?

A. 2
B. 3
C. 4
D. 6

gate2005-it algorithms hashing easy

Answer

Answers: Hashing

1.9.1 Hashing: GATE CSE 1990 | Question: 13b top

https://gateoverflow.in/86225

1.) The worst case can be O(n) if all the insertions collide at the same key.

2.) This happens when we are not using an uniformly distributed hash function or when our hash table has only a single key where all the elements are chained up.

3 votes

-- Aakash Das (511 points)

1.9.2 Hashing: GATE CSE 2020 | Question: 23 top

https://gateoverflow.in/333208

In double hashing scheme, the probe sequence is determined by $(h1(k) + ih2(k)) \mod m$, where i denotes the index in probe sequence and m denotes the hash table size.

Given $h1(k)$ and $h2(k)$, we have to determine the second element of the probe sequence (i.e. $i = 1$) for the key $k = 90$.

$(h1(90) + 1 * h2(90)) \mod 23 = (21 + 15) \mod 23 = 36 \mod 23 = 13$

24 votes

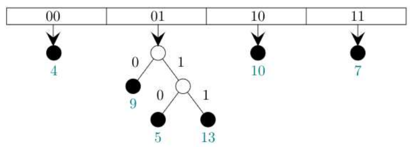
-- Aditya Pradhan (625 points)

1.9.3 Hashing: GATE CSE 2021 Set 1 | Question: 47 top

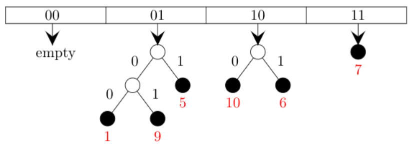
https://gateoverflow.in/357404

- $1-0001$
- $4-0100$
- $5-0101$
- $6-0110$
- $7-0111$
- $9-1001$
- $10-1010$
- $13-1101$
- $14-1110$

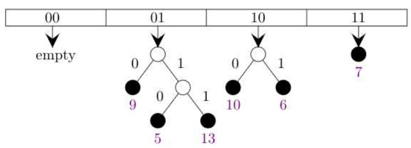
A. $5, 9, 4, 13, 10, 7$



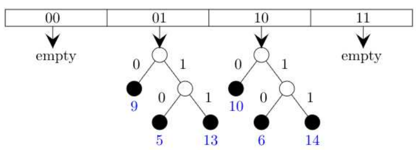
B. $9, 5, 10, 6, 7, 1$



C. $10, 9, 6, 7, 5, 13$



D. $9, 5, 13, 6, 10, 14$



So only one option correct C.

Ans. (C)

5 votes -- Bhargav D Dave (695 points)

1.9.4 Hashing: GATE IT 2005 | Question: 16 top https://gateoverflow.in/3761

✔ 43 in loc 3

165 in loc 5

62 in loc 2

123 in loc 4 (collission and next free space)

142 in loc 6 (collision in 2, and $3, 4, 5$ already occupied)

hence answer **D**

22 votes -- Sankaranarayanan P.N (8.5k points)

1.10 Huffman Code (4) top

1.10.1 Huffman Code: GATE CSE 1989 | Question: 13a top https://gateoverflow.in/93172

A language uses an alphabet of six letters, $\{a, b, c, d, e, f\}$. The relative frequency of use of each letter of the alphabet in the language is as given below:

LETTER	RELATIVE FREQUENCY OF USE
a	0.19
b	0.05
c	0.17
d	0.08
e	0.40
f	0.11

Design a prefix binary code for the language which would minimize the average length of the encoded words of the language.

descriptive gate1989 algorithms huffman-code

Answer

1.10.2 Huffman Code: GATE CSE 2007 | Question: 77 top https://gateoverflow.in/43513

Suppose the letters $a,\ b,\ c,\ d,\ e,\ f$ have probabilities $\dfrac{1}{2}, \dfrac{1}{4}, \dfrac{1}{8}, \dfrac{1}{16}, \dfrac{1}{32}, \dfrac{1}{32}$, respectively.

What is the average length of the Huffman code for the letters $a,\ b,\ c,\ d,\ e,\ f$?

A. 3
B. 2.1875
C. 2.25
D. 1.9375

gate2007-cse algorithms greedy-algorithm normal huffman-code

Answer

1.10.3 Huffman Code: GATE CSE 2017 Set 2 | Question: 50 top https://gateoverflow.in/118395

A message is made up entirely of characters from the set $X = \{P, Q, R, S, T\}$. The table of probabilities for each of the characters is shown below:

Character	Probability
P	0.22
Q	0.34
R	0.17
S	0.19
T	0.08
Total	1.00

If a message of 100 characters over X is encoded using Huffman coding, then the expected length of the encoded message in bits is ______.

gate2017-cse-set2 huffman-code numerical-answers algorithms

Answer

1.10.4 Huffman Code: GATE CSE 2021 Set 2 | Question: 26 top

☛ https://gateoverflow.in/357514

Consider the string abbccddeee. Each letter in the string must be assigned a binary code satisfying the following properties:

1. For any two letters, the code assigned to one letter must not be a prefix of the code assigned to the other letter.
2. For any two letters of the same frequency, the letter which occurs earlier in the dictionary order is assigned a code whose length is at most the length of the code assigned to the other letter.

Among the set of all binary code assignments which satisfy the above two properties, what is the minimum length of the encoded string?

A. 21
B. 23
C. 25
D. 30

gate2021-cse-set2 algorithms huffman-code

Answer

Answers: Huffman Code

1.10.1 Huffman Code: GATE CSE 1989 | Question: 13a top

☛ https://gateoverflow.in/93172

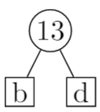
✔ $a-0.19, b-0.05, c-0.17, d-0.08, e-0.40, f-0.11$

Since it is relative frequency we can multiply each with 100 and result remains the same.

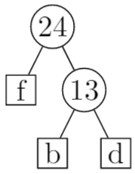
$a-19, b-5, c-17, d-8, e-40, f-11$

For Assigning prefix binary code lets first create the Huffman tree

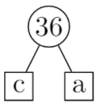
(1) $a-19, b-5, c-17, d-8, e-40, f-11$



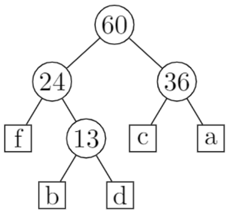
(2) $a-19, (b,d)-13, c-17, e-40, f-11$



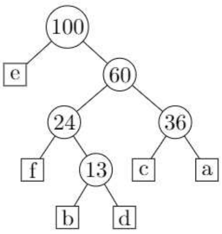
(3) $a - 19, (b, d, f) - 24, c - 17, e - 40$



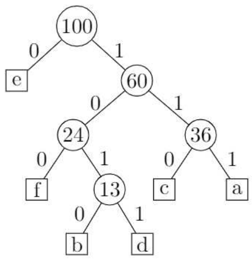
(4) $(a, c) - 36, (b, d, f) - 24, e - 40$



(5) $(a, b, c, d, f) - 60, e - 40$



Now put 0 on each of the left edges and 1 on each of the right edges as shown below



Prefix code = Traverse from the root node to the leaf node and write the symbol $(1\ or\ 0)$ present on each edge.

- For a the prefix code is 111 i.e. 3 bits
- For b the prefix code is 1010 i.e. 4 bits
- For c the prefix code is 110 i.e. 3 bits
- For d the prefix code is 1011 i.e. 4 bits
- For e the prefix code is 0 i.e. 1 bits
- For f the prefix code is 100 i.e. 3 bits

And the average length of encoded words $= \dfrac{(19 \times 3) + (5 \times 4) + (17 \times 3) + (8 \times 4) + (40 \times 1) + (11 \times 3)}{100}$

$$= \frac{57 + 20 + 51 + 32 + 40 + 33}{100}$$

$$= \frac{233}{100}$$

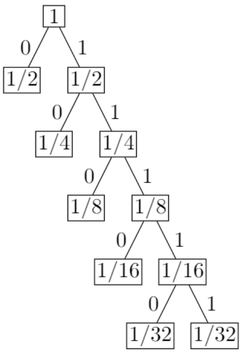
$$= 2.33$$

9 votes -- Satbir Singh (21k points)

1.10.2 Huffman Code: GATE CSE 2007 | Question: 77 top https://gateoverflow.in/43513

✔

Letter	Probability
a	1 / 2
b	1 / 4
c	1 / 8
d	1 / 16
e	1 / 32
f	1 / 32



Prefix Code	Code Length
$a = 0$	1
$b = 10$	2
$c = 110$	3
$d = 1110$	4
$e = 11110$	5
$f = 11111$	5

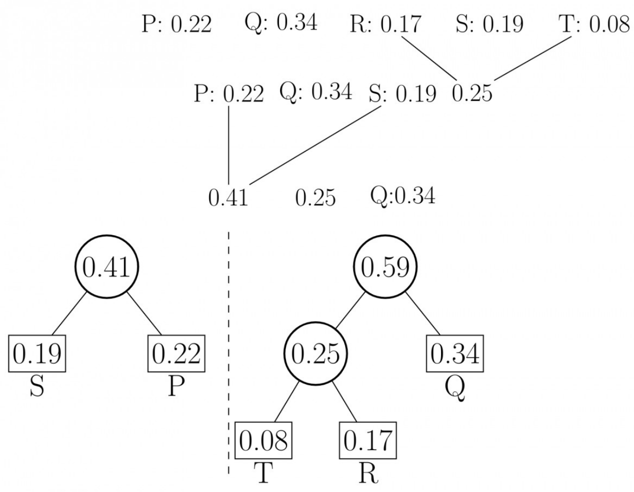
$$\text{Avg length} = \frac{1}{2} \times 1 + \frac{1}{4} \times 2 + \frac{1}{8} \times 3 + \frac{1}{16} \times 4 + \frac{1}{32} \times 5 + \frac{1}{32} \times 5 = \frac{16+16+12+8+5+5}{32} = 1.9375$$

Correct Answer: D

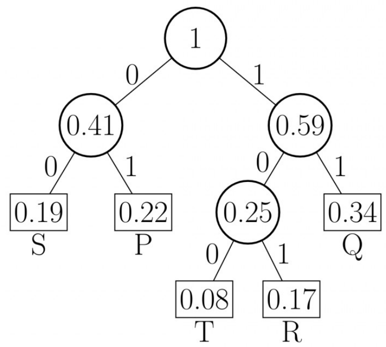
30 votes -- minal (13.1k points)

1.10.3 Huffman Code: GATE CSE 2017 Set 2 | Question: 50 top https://gateoverflow.in/118395

✔ $X = \{P, Q, R, S, T\}$



P:01
Q:11
R:101
S:00
T:100



$\therefore$ Expected length of an encoded character

$$= (0.22 \times 2) + (0.34 \times 2) + (0.17 \times 3) + (0.19 \times 2) + (0.08 \times 3) \text{ bits}$$
$$= 0.44 + 0.68 + 0.51 + 0.38 + 0.24 \text{ bits}$$
$$= 2.25 \text{ bits}$$

$\therefore$ Expected length of a encoded message of 100 characters in bits $= 100 \times 2.25 = 225$

35 votes -- Akash Dinkar (27.9k points)

1.10.4 Huffman Code: GATE CSE 2021 Set 2 | Question: 26 top

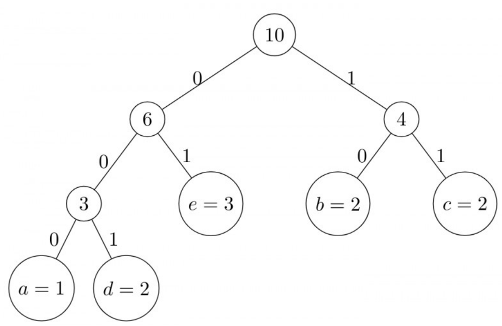
https://gateoverflow.in/357514

✔ Given string: $abbccddeee$

a has the least frequency and should be the leaf of the tree. b, c and d have the same frequency but as per Condition 2 in the questions d should be taken first, followed by c and then b. e has the highest frequency and so must be taken last.

Alphabet	Frequency
a	1
b	2
c	2
d	2
e	3

The final Huffman tree looks like:



Prefix Code	Code Length
$a = 000$	3
$b = 10$	2
$c = 11$	2
$d = 001$	3
$e = 01$	2

$\therefore$ Minimum length of encoded string: $(1 * 3) + (2 * 2) + (2 * 2) + (3 * 2) + (2 * 3) = 23$

Option B is correct.

References:

1. GATE 2007
2. GATE 2017
3. Huffman_coding wiki

References

3 votes -- Hira (13.8k points)

1.11 Identify Function (40) top

1.11.1 Identify Function: GATE CSE 1989 | Question: 8a top https://gateoverflow.in/89080

What is the output produced by the following program, when the input is "HTGATE"

```
Function what (s:string): string;
var n:integer;
begin
      n = s.length
      if n <= 1
      then what := s
      else what :=contact (what (substring (s, 2, n)), s.C [1])
end;
```

Note

i. type string=record
   length:integer;
   C:array[1..100] of char
   end
ii. Substring (s, i, j): this yields the string made up of the i^{th} through j^{th} characters in s; for appropriately defined in i and j.
iii. Contact (s_1, s_2): this function yields a string of length s_1 length + s_2 - length obtained by concatenating s_1 with s_2 such that s_1 precedes s_2.

gate1989 descriptive algorithms identify-function

Answer

1.11.2 Identify Function: GATE CSE 1990 | Question: 11b top https://gateoverflow.in/85987

The following program computes values of a mathematical function $f(x)$. Determine the form of $f(x)$.

```
main ()
{
    int m, n; float x, y, t;
    scanf ("%f%d", &x, &n);
    t = 1; y = 0; m = 1;
    do
    {
        t *= (-x/m);
        y += t;
    } while (m++ < n);
    printf ("The value of y is %f", y);
}
```

gate1990 descriptive algorithms identify-function

Answer

1.11.3 Identify Function: GATE CSE 1991 | Question: 03-viii top

☛ https://gateoverflow.in/523

Consider the following Pascal function:

```
Function X(M:integer):integer;
Var i:integer;
Begin
    i := 0;
    while i*i < M
    do i:= i+1
    X := i
 end
```

The function call $X(N)$, if N is positive, will return

A. $\lfloor\sqrt{N}\rfloor$
B. $\lfloor\sqrt{N}\rfloor+1$
C. $\lceil\sqrt{N}\rceil$
D. $\lceil\sqrt{N}\rceil+1$
E. None of the above

gate1991 algorithms easy identify-function multiple-selects

Answer

1.11.4 Identify Function: GATE CSE 1993 | Question: 7.4 top

☛ https://gateoverflow.in/2292

What does the following code do?

```
var a, b: integer;
begin
    a:=a+b;
    b:=a-b;
    a:a-b;
end;
```

A. exchanges a and b
B. doubles a and stores in b
C. doubles b and stores in a
D. leaves a and b unchanged
E. none of the above

gate1993 algorithms identify-function easy

Answer

1.11.5 Identify Function: GATE CSE 1994 | Question: 6 top

☛ https://gateoverflow.in/2502

What function of x, n is computed by this program?

```
Function what(x, n:integer): integer:
Var
    value : integer
begin
    value := 1
    if n > 0 then
    begin
        if n mod 2 =1 then
            value := value * x;
        value := value * what(x*x, n div 2);
    end;
    what := value;
end;
```

gate1994 algorithms identify-function normal descriptive

Answer

1.11.6 Identify Function: GATE CSE 1995 | Question: 1.4 top https://gateoverflow.in/2591

In the following Pascal program segment, what is the value of X after the execution of the program segment?

```
X := -10; Y := 20;
If X > Y then if X < 0 then X := abs(X) else X := 2*X;
```

A. 10
B. -20
C. -10
D. None

gate1995 algorithms identify-function easy

Answer

1.11.7 Identify Function: GATE CSE 1995 | Question: 2.3 top https://gateoverflow.in/2615

Assume that X and Y are non-zero positive integers. What does the following Pascal program segment do?

```
while X <> Y do
if  X > Y then
    X := X - Y
else
    Y := Y - X;
write(X);
```

A. Computes the LCM of two numbers
B. Divides the larger number by the smaller number
C. Computes the GCD of two numbers
D. None of the above

gate1995 algorithms identify-function normal

Answer

1.11.8 Identify Function: GATE CSE 1995 | Question: 4 top https://gateoverflow.in/2640

A. Consider the following Pascal function where A and B are non-zero positive integers. What is the value of $\text{GET}(3, 2)$?

```
function GET(A,B:integer): integer;
begin
    if B=0 then
        GET:= 1
    else if A < B then
        GET:= 0
    else
        GET:= GET(A-1, B) + GET(A-1, B-1)
end;
```

B. The Pascal procedure given for computing the transpose of an $N \times N, (N > 1)$ matrix A of integers has an error. Find the error and correct it. Assume that the following declaration are made in the main program

```
const
    MAXSIZE=20;
type
    INTARR=array [1..MAXSIZE,1..MAXSIZE] of integer;
Procedure TRANSPOSE (var A: INTARR; N : integer);
var
    I, J, TMP: integer;
begin
    for I:=1 to N - 1 do
    for J:=1 to N do
    begin
        TMP:= A[I, J];
        A[I, J]:= A[J, I];
        A[J, I]:= TMP
    end
end;
```

gate1995 algorithms identify-function normal descriptive

Answer

1.11.9 Identify Function: GATE CSE 1998 | Question: 2.12 top

https://gateoverflow.in/1684

What value would the following function return for the input $x = 95$?

```
Function fun (x:integer):integer;
Begin
    If x > 100 then fun = x - 10
    Else fun = fun(fun (x+11))
End;
```

A. 89
B. 90
C. 91
D. 92

gate1998 algorithms recursion identify-function normal

Answer

1.11.10 Identify Function: GATE CSE 1999 | Question: 2.24 top

https://gateoverflow.in/1501

Consider the following C function definition

```
int Trial (int a, int b, int c)
{
    if ((a>=b) && (c<b)) return b;
    else if (a>=b) return Trial(a, c, b);
    else return Trial(b, a, c);
}
```

The functional Trial:

A. Finds the maximum of a, b, and c

B. Finds the minimum of a, b, and c

C. Finds the middle number of a, b, c

D. None of the above

gate1999 algorithms identify-function normal

Answer

1.11.11 Identify Function: GATE CSE 2000 | Question: 2.15 top

https://gateoverflow.in/662

Suppose you are given an array $s[1....n]$ and a procedure reverse (s, i, j) which reverses the order of elements in s between positions i and j (both inclusive). What does the following sequence do, where $1 \leqslant k \leqslant n$:

```
        reverse (s, 1, k);
        reverse (s, k+1, n);
        reverse (s, 1, n);
```

A. Rotates s left by k positions
B. Leaves s unchanged
C. Reverses all elements of s
D. None of the above

gate2000-cse algorithms normal identify-function

Answer

1.11.12 Identify Function: GATE CSE 2003 | Question: 1 top

https://gateoverflow.in/892

Consider the following C function.

For large values of y, the return value of the function f best approximates

```
float f,(float x, int y) {
    float p, s; int i;
    for (s=1,p=1,i=1; i<y; i++) {
        p *= x/i;
        s += p;
```

Remember e^x series

```
    }
    return s;
}
```

A. x^y
B. e^x
C. $\ln(1+x)$
D. x^x

gate2003-cse algorithms identify-function normal

Answer

1.11.13 Identify Function: GATE CSE 2003 | Question: 88 top

https://gateoverflow.in/971

In the following C program fragment, j, k, n and TwoLog_n are integer variables, and A is an array of integers. The variable n is initialized to an integer $\geqslant 3$, and TwoLog_n is initialized to the value of $2^* \lceil \log_2(n) \rceil$

```
for (k = 3;  k <= n; k++)
        A[k] = 0;
for (k = 2; k <= TwoLog_n; k++)
    for (j = k+1; j <= n; j++)
        A[j] = A[j] || (j%k);
for (j = 3; j <= n; j++)
    if (!A[j]) printf("%d", j);
```

The set of numbers printed by this program fragment is

A. $\{m \mid m \leq n, (\exists i)\ [m = i!]\}$
B. $\{m \mid m \leq n, (\exists i)\ [m = i^2]\}$
C. $\{m \mid m \leq n, \text{m is prime}\}$
D. $\{\ \}$

gate2003-cse algorithms identify-function normal

Answer

1.11.14 Identify Function: GATE CSE 2004 | Question: 41 top

https://gateoverflow.in/1038

Consider the following C program

```
main()
{
    int x, y, m, n;
    scanf("%d %d", &x, &y);
    /* Assume x>0 and y>0*/
    m = x; n = y;
    while(m != n)
        {
            if (m > n)
                m = m-n;
            else
                n = n-m;
        }
    printf("%d", n);
}
```

The program computes

A. $x + y$ using repeated subtraction
B. $x \mod y$ using repeated subtraction
C. the greatest common divisor of x and y
D. the least common multiple of x and y

gate2004-cse algorithms normal identify-function

Answer

1.11.15 Identify Function: GATE CSE 2004 | Question: 42 top https://gateoverflow.in/1039

What does the following algorithm approximate? (Assume $m > 1, \epsilon > 0$).

```
x = m;
y = 1;
While (x-y > ϵ)
{
    x = (x+y)/2;
    y = m/x;
}
print(x);
```

A. $\log m$
B. m^2
C. $m^{\frac{1}{2}}$
D. $m^{\frac{1}{3}}$

gate2004-cse algorithms identify-function normal

Answer

1.11.16 Identify Function: GATE CSE 2005 | Question: 31 top https://gateoverflow.in/1367

Consider the following C-program:

```
void foo (int n, int sum) {
    int k = 0, j = 0;
    if (n == 0) return;
    k = n % 10; j = n/10;
    sum = sum + k;
    foo (j, sum);
    printf ("%d,",k);
}

int main() {
    int a = 2048, sum = 0;
    foo(a, sum);
    printf("%d\n", sum);
}
```

What does the above program print?

A. 8, 4, 0, 2, 14
B. 8, 4, 0, 2, 0
C. 2, 0, 4, 8, 14
D. 2, 0, 4, 8, 0

gate2005-cse algorithms identify-function recursion normal

Answer

1.11.17 Identify Function: GATE CSE 2006 | Question: 50 top https://gateoverflow.in/1828

A set X can be represented by an array $x[n]$ as follows:

$$x\left[i\right] = \begin{cases} 1 & \text{if } i \in X \\ 0 & \text{otherwise} \end{cases}$$

Consider the following algorithm in which x, y, and z are Boolean arrays of size n:

```
algorithm zzz(x[], y[], z[]) {
    int i;

    for(i=0; i<n; ++i)
        z[i] = (x[i] ∧ ~y[i]) ∨ (~x[i] ∧ y[i]);
}
```

The set Z computed by the algorithm is:

A. $(X \cup Y)$
B. $(X \cap Y)$
C. $(X - Y) \cap (Y - X)$
D. $(X - Y) \cup (Y - X)$

gate2006-cse algorithms identify-function normal

Answer

1.11.18 Identify Function: GATE CSE 2006 | Question: 53 top

https://gateoverflow.in/1831

Consider the following C-function in which $a[n]$ and $b[m]$ are two sorted integer arrays and $c[n+m]$ be another integer array,

```
void xyz(int a[], int b [], int c []){
    int i,j,k;
    i=j=k=0;
    while ((i<n) && (j<m))
        if (a[i] < b[j]) c[k++] = a[i++];
        else c[k++] = b[j++];
}
```

Which of the following condition(s) hold(s) after the termination of the while loop?

i. $j < m, k = n + j - 1$ and $a[n-1] < b[j]$ if $i = n$
ii. $i < n, k = m + i - 1$ and $b[m-1] \leq a[i]$ if $j = m$

A. only (i)
B. only (ii)
C. either (i) or (ii) but not both
D. neither (i) nor (ii)

gate2006-cse algorithms identify-function normal

Answer

1.11.19 Identify Function: GATE CSE 2009 | Question: 18 top

https://gateoverflow.in/1310

Consider the program below:

```
#include <stdio.h>
int fun(int n, int *f_p) {
    int t, f;
    if (n <= 1) {
        *f_p = 1;
        return 1;
    }
    t = fun(n-1, f_p);
    f = t + *f_p;
    *f_p = t;
    return f;
}

int main() {
    int x = 15;
    printf("%d/n", fun(5, &x));
    return 0;
}
```

The value printed is:

A. 6
B. 8
C. 14
D. 15

gate2009-cse algorithms recursion identify-function normal

Answer

1.11.20 Identify Function: GATE CSE 2010 | Question: 35 top

https://gateoverflow.in/2336

What is the value printed by the following C program?

```
#include<stdio.h>

int f(int *a, int n)
{
```

```
    if (n <= 0) return 0;
    else if (*a % 2 == 0) return *a+f(a+1, n-1);
    else return *a - f(a+1, n-1);
}

int main()
{
    int a[] = (12, 7, 13, 4, 11, 6);
    printf("%d", f(a, 6));
    return 0;
}
```

A. -9
B. 5
C. 15
D. 19

gate2010-cse algorithms recursion identify-function normal

Answer

1.11.21 Identify Function: GATE CSE 2011 | Question: 48 top

https://gateoverflow.in/2154

Consider the following recursive C function that takes two arguments.

```
unsigned int foo(unsigned int n, unsigned int r) {
    if (n>0) return ((n%r) + foo(n/r, r));
    else return 0;
}
```

What is the return value of the function foo when it is called as $\text{foo}(345, 10)$?

A. 345
B. 12
C. 5
D. 3

gate2011-cse algorithms recursion identify-function normal

Answer

1.11.22 Identify Function: GATE CSE 2011 | Question: 49 top

https://gateoverflow.in/43324

Consider the following recursive C function that takes two arguments.

```
unsigned int foo(unsigned int n, unsigned int r) {
    if (n>0) return ((n%r) + foo(n/r, r));
    else return 0;
}
```

What is the return value of the function foo when it is called as $\text{foo}(513, 2)$?

A. 9
B. 8
C. 5
D. 2

gate2011-cse algorithms recursion identify-function normal

Answer

1.11.23 Identify Function: GATE CSE 2013 | Question: 31 top

https://gateoverflow.in/1542

Consider the following function:

```
int unknown(int n){

int i, j, k=0;
for (i=n/2; i<=n; i++)
    for (j=2; j<=n; j=j*2)
        k = k + n/2;
 return (k);

}
```

The return value of the function is

A. $\Theta(n^2)$
B. $\Theta(n^2 \log n)$
C. $\Theta(n^3)$
D. $\Theta(n^3 \log n)$

gate2013-cse algorithms identify-function normal

Answer

1.11.24 Identify Function: GATE CSE 2014 Set 1 | Question: 41 top https://gateoverflow.in/1919

Consider the following C function in which **size** is the number of elements in the array **E**:

```
int MyX(int *E, unsigned int size)
{
   int Y = 0;
   int Z;
   int i, j, k;

   for(i = 0; i< size; i++)
          Y = Y + E[i];

    for(i=0; i < size; i++)
        for(j = i; j < size; j++)
        {
            Z = 0;
            for(k = i; k <= j; k++)
               Z = Z + E[k];
            if(Z > Y)
               Y = Z;
        }
   return Y;
}
```

The value returned by the function **MyX** is the

A. maximum possible sum of elements in any sub-array of array **E**.
B. maximum element in any sub-array of array **E.**
C. sum of the maximum elements in all possible sub-arrays of array **E**.
D. the sum of all the elements in the array **E.**

gate2014-cse-set1 algorithms identify-function normal

Answer

1.11.25 Identify Function: GATE CSE 2014 Set 2 | Question: 10 top https://gateoverflow.in/1964

Consider the function func shown below:

```
int func(int num) {
   int count = 0;
   while (num) {
     count++;
     num>>= 1;
   }
   return (count);
}
```

The value returned by func(435) is ________

gate2014-cse-set2 algorithms identify-function numerical-answers easy

Answer

1.11.26 Identify Function: GATE CSE 2014 Set 3 | Question: 10 top https://gateoverflow.in/2044

Let A be the square matrix of size $n \times n$. Consider the following pseudocode. What is the expected output?

```
C=100;
for i=1 to n do
    for j=1 to n do
    {
        Temp = A[i][j]+C;
```

```
        A[i][j] = A[j][i];
        A[j][i] = Temp -C;
    }
for i=1 to n do
    for j=1 to n do
        output (A[i][j]);
```

A. The matrix A itself
B. Transpose of the matrix A
C. Adding 100 to the upper diagonal elements and subtracting 100 from lower diagonal elements of A
D. None of the above

gate2014-cse-set3 algorithms identify-function easy

Answer

1.11.27 Identify Function: GATE CSE 2015 Set 1 | Question: 31 top https://gateoverflow.in/8263

Consider the following C function.

```
int fun1 (int n) {
     int i, j, k, p, q = 0;
     for (i = 1; i < n; ++i)
     {
        p = 0;
       for (j = n; j > 1; j = j/2)
           ++p;
       for (k = 1; k < p; k = k * 2)
           ++q;
     }
     return q;
}
```

Which one of the following most closely approximates the return value of the function fun1?

A. n^3
B. $n(\log n)^2$
C. $n \log n$
D. $n \log(\log n)$

gate2015-cse-set1 algorithms normal identify-function

Answer

1.11.28 Identify Function: GATE CSE 2015 Set 2 | Question: 11 top https://gateoverflow.in/8060

Consider the following C function.

```
int fun(int n) {
    int x=1, k;
    if (n==1) return x;
    for (k=1; k<n; ++k)
        x = x + fun(k) * fun (n-k);
    return x;
}
```

The return value of $fun(5)$ is ______.

gate2015-cse-set2 algorithms identify-function recurrence normal numerical-answers

Answer

1.11.29 Identify Function: GATE CSE 2015 Set 3 | Question: 49 top https://gateoverflow.in/8558

Suppose $c = \langle c[0], \dots, c[k-1] \rangle$ is an array of length k, where all the entries are from the set $\{0, 1\}$. For any positive integers a and n, consider the following pseudocode.

DOSOMETHING (*c, a, n*)

$z \leftarrow 1$
for $i \leftarrow 0$ to $k-1$
    do $z \leftarrow z^2 \bmod n$
    if $c[i] = 1$
        then $z \leftarrow (z \times a) \bmod n$

return z

If $k = 4, c = \langle 1, 0, 1, 1 \rangle, a = 2,$ and $n = 8$, then the output of DOSOMETHING(*c, a, n)* is _______.

gate2015-cse-set3 algorithms identify-function normal numerical-answers

Answer

1.11.30 Identify Function: GATE CSE 2017 Set 2 | Question: 14 top https://gateoverflow.in/118245

Consider the following function implemented in C:

```
void printxy(int x, int y) {
    int *ptr;
    x=0;
    ptr=&x;
    y=*ptr;
    *ptr=1;
    printf("%d, %d", x, y);
}
```

The output of invoking $printxy(1, 1)$ is:

A. $0, 0$
B. $0, 1$
C. $1, 0$
D. $1, 1$

gate2017-cse-set2 identify-function

Answer

1.11.31 Identify Function: GATE CSE 2019 | Question: 18 top https://gateoverflow.in/302830

Consider the following C program :

```
#include<stdio.h>
int jumble(int x, int y){
    x = 2*x+y;
    return x;
}
int main(){
    int x=2, y=5;
    y=jumble(y,x);
    x=jumble(y,x);
    printf("%d \n",x);
    return 0;
}
```

The value printed by the program is ______________.

gate2019-cse numerical-answers identify-function

Answer

1.11.32 Identify Function: GATE CSE 2019 | Question: 26 top https://gateoverflow.in/302822

Consider the following C function.

```
void convert (int n ) {
        if (n<0)
            printf("%d", n);
        else {
            convert(n/2);
            printf("%d", n%2);
        }
}
```

Which one of the following will happen when the function *convert* is called with any positive integer n as argument?

A. It will print the binary representation of n and terminate
B. It will print the binary representation of n in the reverse order and terminate
C. It will print the binary representation of n but will not terminate
D. It will not print anything and will not terminate

gate2019-cse algorithms identify-function

Answer

1.11.33 Identify Function: GATE CSE 2020 | Question: 48 top

https://gateoverflow.in/333183

Consider the following C functions.

```
int tob (int b, int* arr) {
    int i;
    for (i = 0; b>0; i++)   {
        if (b%2)  arr [i] = 1;
        else      arr[i] = 0;
        b = b/2;
    }
    return (i);
}
```

```
int pp(int a, int b)   {
    int  arr[20];
    int i, tot = 1, ex, len;
    ex = a;
    len = tob(b, arr);
    for (i=0; i<len ; i++) {
         if (arr[i] ==1)
             tot = tot * ex;
         ex= ex*ex;
    }
return (tot) ;
}
```

The value returned by $pp(3,4)$ is _______.

gate2020-cse numerical-answers identify-function

Answer

1.11.34 Identify Function: GATE CSE 2021 Set 1 | Question: 48 top

https://gateoverflow.in/357403

Consider the following ANSI C function:

```
int SimpleFunction(int Y[], int n, int x)
{
int total = Y[0], loopIndex;
for (loopIndex=1; loopIndex<=n-1; loopIndex++)
    total=x*total +Y[loopIndex];
return total;
}
```

Let Z be an array of 10 elements with $\mathsf{Z}[i] = 1$, for all i such that $0 \leq i \leq 9$. The value returned by $\mathsf{SimpleFunction}(\mathsf{Z}, 10, 2)$ is __________

gate2021-cse-set1 algorithms numerical-answers identify-function

Answer

1.11.35 Identify Function: GATE CSE 2021 Set 2 | Question: 23 top

https://gateoverflow.in/357517

Consider the following ANSI C function:

```
int SomeFunction (int x, int y)
{
    if ((x==1) || (y==1)) return 1;
    if (x==y) return x;
    if (x > y) return SomeFunction(x-y, y);
    if (y > x) return SomeFunction (x, y-x);

}
```

The value returned by $\text{SomeFunction}(15, 255)$ is __________

gate2021-cse-set2 numerical-answers algorithms identify-function output

Answer

1.11.36 Identify Function: GATE IT 2005 | Question: 53 top

https://gateoverflow.in/3814

The following C function takes two ASCII strings and determines whether one is an anagram of the other. An anagram of a string s is a string obtained by permuting the letters in s.

```
int anagram (char *a, char *b) {
    int count [128], j;
    for (j = 0;  j < 128; j++) count[j] = 0;
    j = 0;
    while (a[j] && b[j]) {
```

```
        A;
        B;
    }
    for (j = 0; j < 128; j++) if (count [j]) return 0;
    return 1;
}
```

Choose the correct alternative for statements A and B.

A. A: count [a[j]]++ and B: count[b[j]]--
B. A: count [a[j]]++ and B: count[b[j]]++
C. A: count [a[j++]]++ and B: count[b[j]]--
D. A: count [a[j]]++ and B: count[b[j++]]--

gate2005-it normal identify-function

Answer

1.11.37 Identify Function: GATE IT 2005 | Question: 57 top

https://gateoverflow.in/3818

What is the output printed by the following program?

```
#include <stdio.h>

int f(int n, int k) {
    if (n == 0) return 0;
    else if (n % 2) return f(n/2, 2*k) + k;
    else return f(n/2, 2*k) - k;
}

int main () {
    printf("%d", f(20, 1));
    return 0;
}
```

A. 5
B. 8
C. 9
D. 20

gate2005-it algorithms identify-function normal

Answer

1.11.38 Identify Function: GATE IT 2006 | Question: 52 top

https://gateoverflow.in/3595

The following function computes the value of $\binom{m}{n}$ correctly for all legal values m and n $(m \geq 1, n \geq 0$ and $m > n)$

```
int func(int m, int n)
{
    if (E) return 1;
    else return(func(m -1, n) + func(m - 1, n - 1));
}
```

In the above function, which of the following is the correct expression for E?

A. $(n == 0) || (m == 1)$
B. $(n == 0)$ && $(m == 1)$
C. $(n == 0) || (m == n)$
D. $(n == 0)$ && $(m == n)$

gate2006-it algorithms identify-function normal

Answer

1.11.39 Identify Function: GATE IT 2008 | Question: 82 top

https://gateoverflow.in/3406

Consider the code fragment written in C below :

```
void f (int n)
{
  if (n <=1)  {
```

```
    printf ("%d", n);
  }
  else {
    f (n/2);
    printf ("%d", n%2);
  }
}
```

What does f(173) print?

A. 010110101
B. 010101101
C. 10110101
D. 10101101

gate2008-it algorithms recursion identify-function normal

Answer

1.11.40 Identify Function: GATE IT 2008 | Question: 83 top

https://gateoverflow.in/3407

Consider the code fragment written in C below :

```
void f (int n)
{
    if (n <= 1)  {
        printf ("%d", n);
    }
    else {
        f (n/2);
        printf ("%d", n%2);
    }
}
```

Which of the following implementations will produce the same output for $f(173)$ as the above code?

P1

```
void f (int n)
{
    if (n/2)  {
        f(n/2);
    }
    printf ("%d", n%2);
}
```

P2

```
void f (int n)
{
    if (n <=1)  {
        printf ("%d", n);
    }
    else {
        printf ("%d", n%2);
        f (n/2);
    }
}
```

A. Both $P1$ and $P2$
B. $P2$ only
C. $P1$ only
D. Neither $P1$ nor $P2$

gate2008-it algorithms recursion identify-function normal

Answer

Answers: Identify Function

1.11.1 Identify Function: GATE CSE 1989 | Question: 8a top

https://gateoverflow.in/89080

✔ This function is reversing the string "HTGATE".

Here, function substring() gives the substring from the 2^{nd} character of the original string till the end and contact(p,q) is concatenating the strings p and q.

$n = s.$length means, $n =$ length of the string and $c[1]$ means, 1^{st} character of the passed string

So, it will work like this:

- what(HTGATE)
- contact(what(TGATE),H)

- contact(contact(what(GATE),T),H)
- contact(contact(contact(what(ATE),G),T),H)
- contact(contact(contact(contact(what(TE),A),G),T),H)
- contact(contact(contact(contact(contact(what(E),T),A),G),T),H)
- contact(contact(contact(contact(contact(E,T),A),G),T),H)
- contact(contact(contact(contact(ET,A),G),T),H)
- contact(contact(contact(ETA,G),T),H)
- contact(contact(ETAG,T),H)
- contact(ETAGT,H)

ETAGTH

9 votes -- ankitgupta.1729 (15k points)

1.11.2 Identify Function: GATE CSE 1990 | Question: 11b top

https://gateoverflow.in/85987

✔ while(m++ < n)

here , we have used post-increment operator for variable m. So, first we will check whether $m < n$ is true or not then we will increase the value of 'm' as $m = m + 1$.

So, for $n = 2$, first we will go in the loop.

So, $t = 1 * (\frac{-x}{1})$ and $y = 0 + (1 * (\frac{-x}{1})) = \frac{-x}{1}$

Now, we will check the condition as :- $(1 < 2)$. Since, it is true so we will go in the loop again as well as we will increment the value of m. So, now, $m = 2$

Now again, in the body of the loop,

$t = (1 * (\frac{-x}{1})) * (\frac{-x}{2})$ and $y = (1 * (\frac{-x}{1})) + (1 * (\frac{-x}{1}) * (\frac{-x}{2})) = (-1)^1 * \frac{x^1}{1} + (-1)^2 * \frac{x^2}{1*2}$

Like this when we do then at certain point of time , we will check $((n-1) < n)$ or not for the value of $m = n - 1$. since it is true . So, again we will go in the loop and increment in value of m.So, now , $m = n$

So, Now , $y = 0 + \frac{(-1)^1 * (x^1)}{1} + \frac{(-1)^2 * x^2}{1*2} + \frac{(-1)^3 * x^3}{1*2*3} + \dots + \frac{(-1)^n * x^n}{1*2*3*\dots*n} = \sum_{i=1}^{n} \frac{(-1)^i * x^i}{i!}$

Now again, we will check the condition. Since $(n < n)$ is false. So, we will not go in loop again and print the value of y which is $\sum_{i=1}^{n} \frac{(-1)^i * x^i}{i!}$

10 votes -- ankitgupta.1729 (15k points)

1.11.3 Identify Function: GATE CSE 1991 | Question: 03-viii top

https://gateoverflow.in/523

✔ For $N = 9$, it returns 3.

For $N = 10$ it returns 4.

For $N = 16$ it returns 4.

For $N = 17$ it returns 5.

So answer should be **C**.

17 votes -- Taymiyyah Bhat (2.3k points)

1.11.4 Identify Function: GATE CSE 1993 | Question: 7.4 top

https://gateoverflow.in/2292

✔ Answer is simply A i.e. it swaps the values of the two.. Take any two values for A and B. and perform the given operations over them.

25 votes -- Gate Keeda (15.9k points)

1.11.5 Identify Function: GATE CSE 1994 | Question: 6 top

https://gateoverflow.in/2502

✔ answer - x^n

14 votes -- Ankit Rokde (6.9k points)

1.11.6 Identify Function: GATE CSE 1995 | Question: 1.4 top https://gateoverflow.in/2591

✔ The answer of X remains unchanged. As the if condition becomes false.

```
X := -10
```

The answer is **C** . This is a classic example of an if-else issue. Always $else$ matches for nesting to the closest if in C Programming & Pascal .
https://en.wikipedia.org/wiki/Dangling_else

```
if (x>y)
{
   if (x<0)
       x=abs(x)
    else
       x=2*x
}
```

References

30 votes -- Akash Kanase (36k points)

1.11.7 Identify Function: GATE CSE 1995 | Question: 2.3 top https://gateoverflow.in/2615

✔ Answer: **C**

Let $X = 3$ and $Y = 7$.

- 1^{st} pass: $X = 3, Y = 4$
- 2^{nd} pass: $X = 3, Y = 1$
- 3^{rd} pass: $X = 2, Y = 1$
- 4^{th} pass: $X = 1, Y = 1$

write (X), which writes 1.

Ref: https://en.wikipedia.org/wiki/Euclidean_algorithm

References

17 votes -- Rajarshi Sarkar (27.8k points)

1.11.8 Identify Function: GATE CSE 1995 | Question: 4 top https://gateoverflow.in/2640

✔
A. 3
B.

```
begin
    for I:=2 to N do
    for J:=1 to ( I-1) do
    begin
        TMP:= A[I, J];
        A[I, J]:= A[J, I];
        A[J, I]:= TMP
    end
end
```

Should be the condition.

12 votes -- papesh (18k points)

1.11.9 Identify Function: GATE CSE 1998 | Question: 2.12 top https://gateoverflow.in/1684

✔ Value returned by $\text{fun}(95) = \text{fun}(\text{fun}(106))$

$$\begin{aligned}
&= \text{fun}(96)\\
&= \text{fun}(\text{fun}(107))\\
&= \text{fun}(97)\\
&= \text{fun}(\text{fun}(108))\\
&= \text{fun}(98)\\
&= \text{fun}(\text{fun}(109))\\
&= \text{fun}(99)\\
&= \text{fun}(\text{fun}(110))\\
&= \text{fun}(100)\\
&= \text{fun}(\text{fun}(111))\\
&= \text{fun}(101) = 91.
\end{aligned}$$

Correct Answer: C

👍 36 votes -- Digvijay (44.9k points)

1.11.10 Identify Function: GATE CSE 1999 | Question: 2.24 top ☛ https://gateoverflow.in/1501

✔

a	b	c	Return
1	1	1	The final return statement is $c < b$. So, this never returns.

Answer D.

👍 67 votes -- Arjun Suresh (330k points)

1.11.11 Identify Function: GATE CSE 2000 | Question: 2.15 top ☛ https://gateoverflow.in/662

✔ Answer is **A.**

Effect of the above 3 reversals for any K is equivalent to left rotation of the array of size n by k.

Let , $S[1\dots7] = \boxed{1}\boxed{2}\boxed{3}\boxed{4}\boxed{5}\boxed{6}\boxed{7}$

So, $n = 7, k = 2$

reverse $(S, 1, 2)$ we get $[2, 1, 3, 4, 5, 6, 7]$

reverse $(S, 3, 7)$ we get $[2, 1, 7, 6, 5, 4, 3]$

reverse $(S, 1, 7)$ we get $[3, 4, 5, 6, 7, 1, 2]$

Hence, option (A) rotates s left by k positions and is correct.

👍 46 votes -- Kalpna Bhargav (2.5k points)

1.11.12 Identify Function: GATE CSE 2003 | Question: 1 top ☛ https://gateoverflow.in/892

✔ A simplified version of the given program can be written as:

```
float f(float x, int y) {
    float p=1, s=1;
    int i;
    for (i=1; i<y; i++) {
        p = p * (x/i);
        s = s + p;
    }
    return s;
}
```

We can take $p = 1, s = 1$ initialization outside of for loop because there is no condition checking in for loop involving p, s .

i	$p = p^*(x/i)$	$s = s + p$
1	x	$1 + x$
2	$\frac{x^2}{2}$	$1 + x + \frac{x^2}{2}$
3	$\frac{x^3}{6}$	$1 + x + \frac{x^2}{2} + \frac{x^3}{6}$
4	$\frac{x^4}{24}$	$1 + x + \frac{x^2}{2} + \frac{x^3}{6} + \frac{x^4}{24}$
n	$\frac{x^n}{n!}$	e^x

$$\boxed{e^x = \sum_{n=0}^{\infty} \frac{x^n}{n!} = 1 + x + \frac{x^2}{2} + \frac{x^3}{6} + \frac{x^4}{24} + \ldots + \frac{x^n}{n!}}$$

Hence, option **B** is answer.

40 votes -- Rajesh Pradhan (18.9k points)

1.11.13 Identify Function: GATE CSE 2003 | Question: 88 top

https://gateoverflow.in/971

✔ The nested loop is taking all integers from 2 to $2 * \log_2 n$. Take all their non-multiples before n, and make the corresponding entry in A as 1. For example, for 2, and $n = 10$, $A[3], A[5], A[7]$, and $A[9]$are made 1. Similarly for 3, 4, ... till $2 * \log n$. So, if any entry $A[p]$ is 1 means it must be a multiple of $2, 3, \ldots. 2 * \log_2 n$, which is $(2 \log n)!$ and is greater than n. So, for no index p, $A[p]$ will be 0. So, answer is **D**.

Suppose the line

```
A[j] = A[j] || (j%k);
```

is replaced with

```
A[j] = A[j] || !(j%k);
```

Now, the nested loop is taking all integers from 2 to $\log_2 n$, take all their multiples before n, and make the corresponding entry in A as 1. For example, for 2, and $n = 10$, $A[4], A[6], A[8]$ and $A[10]$ are made 1. Similarly for $3, 4, \ldots$ till $2 * \log n$. So, for all non-prime indices of A, we will have a 1, and for prime indices we have a 0. And we print i if $A[j]$ is 0 meaning j is prime.

32 votes -- Arjun Suresh (330k points)

1.11.14 Identify Function: GATE CSE 2004 | Question: 41 top

https://gateoverflow.in/1038

✔ It is an algorithm for $\gcd$ computation.

Here, while loop executes until $m = n$.
We can test by taking any two numbers as m, n.

Answer will be (**C**).

Ref: https://www.khanacademy.org/computing/computer-science/cryptography/modarithmetic/a/the-euclidean-algorithm

References

13 votes -- srestha (85.2k points)

1.11.15 Identify Function: GATE CSE 2004 | Question: 42 top

https://gateoverflow.in/1039

✔ By putting $y = m/x$ into $x = (x + y)/2$
$x = (x + m/x)/2$

$\implies 2x^2 = x^2 + m$
$\implies x = m^{1/2}$

We can also check by putting 2 or 3 different values also.

Correct Answer: C

56 votes -- gate_asp (615 points)

1.11.16 Identify Function: GATE CSE 2005 | Question: 31 top https://gateoverflow.in/1367

✔ Correct Option: **D**

foo is printing the lowest digit. But the $printf$ inside it is after the recursive call. This forces the output to be in reverse order

$2, 0, 4, 8$

The final value sum printed will be 0 as C uses pass by value and hence the modified value inside foo won't be visible inside $main$.

25 votes -- anshu (2.7k points)

1.11.17 Identify Function: GATE CSE 2006 | Question: 50 top https://gateoverflow.in/1828

✔ Correct Option: **D**

In the given algorithm the for loop contains a logical expression

```
z[i] = (x[i] ∧ ~y[i]) ∨ (~x[i] ∧ y[i]);
```

The equivalent set representation of a given logical expression if we assume $z[i] = Z, x[i] = X, y[i] = Y$ then

$Z = (X \wedge \neg Y) \vee (\neg X \wedge Y)$

$\implies Z = (X - Y) \cup (Y - X) [\because A \wedge \neg B = A - B]$

24 votes -- Prasanna Ranganathan (3.9k points)

1.11.18 Identify Function: GATE CSE 2006 | Question: 53 top https://gateoverflow.in/1831

✔ The while loop adds elements from a and b (whichever is smaller) to c and terminates when either of them exhausts. So, when loop terminates either $i = n$ or $j = m$.

Suppose $i = n$. This would mean all elements from array a are added to $c =>$ k must be incremented by n. c would also contain j elements from array b. So, number of elements in c would be $n + j$ and hence $k = n + j$.

Similarly, when $j = m$, $k = m + i$.

Hence, option (**D**) is correct. (Had k started from -1 and not 0 and we used $++k$ inside loop, answer would have been option (C))

71 votes -- Arjun Suresh (330k points)

1.11.19 Identify Function: GATE CSE 2009 | Question: 18 top https://gateoverflow.in/1310

✔ The answer is B.

Let the address of x be 1000.

1. $f(5, 1000) = 8$
2. $f(4, 1000) = 5$
3. $f(3, 1000) = 3$
4. $f(2, 1000) = 2$
5. $f(1, 1000) = 1$.

The evaluation is done from 5 to 1. Since recursion is used.

18 votes -- Gate Keeda (15.9k points)

1.11.20 Identify Function: GATE CSE 2010 | Question: 35 top https://gateoverflow.in/2336

✔ Suppose int array takes 4 bytes for each element and stored at base address 100.

Follow below image. Red color shows the return value.

So, 15 is the answer.

Correct Answer: C

👍 18 votes -- Rajesh Pradhan (18.9k points)

It will print

$$12 + (7 - (13 - (4 + (11 - (6 + 0)))))$$
$$= 12 + (7 - (13 - (4 + (11 - 6))))$$
$$= 12 + 7 - 13 + 9$$
$$= 15$$

👍 26 votes -- gatecse (62.6k points)

1.11.21 Identify Function: GATE CSE 2011 | Question: 48 top

https://gateoverflow.in/2154

✔ Red colour represents return values.

Answer is **12.**

$$\overset{12}{\text{foo}(345, 10)}$$
$$\Downarrow$$
$$5 + \overset{7}{\text{foo}(34, 10)}$$
$$\Downarrow$$
$$4 + \overset{3}{\text{foo}(3, 10)}$$
$$\Downarrow$$
$$3 + \text{foo}(0, 10)$$
$$\Downarrow$$
$$0$$

👍 22 votes -- Rajesh Pradhan (18.9k points)

1.11.22 Identify Function: GATE CSE 2011 | Question: 49 top

https://gateoverflow.in/43324

✔ The function returns the sum of digits in a binary representation of the given number

so $1 + 0 + 0 + 0 + 0 + 0 + 0 + 0 + 0 + 1 = 2$

Correct Answer: D

👍 25 votes -- Sandeep_Uniyal (6.5k points)

1.11.23 Identify Function: GATE CSE 2013 | Question: 31 top

https://gateoverflow.in/1542

✔ The outer loop is running for $n/2$ times and inner loop is running for $\log_2 n$ times (each iteration doubles j and j stops at n means $\log_2 n$ times j loop will iterate).

Now in each iteration k is incremented by $n/2$. So, overall k will be added $n/2 * \log n * n/2$ with an initial value of 0. So, final value of k will be $\Theta(n^2 \log n)$.

Correct Answer: B.

👍 70 votes -- Arjun Suresh (330k points)

1.11.24 Identify Function: GATE CSE 2014 Set 1 | Question: 41 top

https://gateoverflow.in/1919

✔ Answer is (**A**) maximum possible sum of elements in any sub-array of array **E**.

```
int MyX ( int * E, unsigned int size )
{
    int Y= 0;
    int z;
    int i, j,k;

//calculate sum of the elements of the array E and stores it in Y
    for i 0;i<size;i++)
      Y = Y+E[i];
```

```
//calculate the sum of all possible subaarays
//(starting from postion 0..n-1)
    for (i=0;i<size;i++)
        for(j=i;j<size ;j++)
        {
            z = 0;
            for(k=i; k<=j;k++)
                z=z+E[k];

//checks whether sum of elements of each subarray is greater
//than the current max, if so, then assign it to currentmax
            if(z>Y)
                Y = z;
        }
//ultimately returns the maximum possible sum of elements
//in any sub array of given array E
    return Y;
}
```

42 votes -- Kalpna Bhargav (2.5k points)

1.11.25 Identify Function: GATE CSE 2014 Set 2 | Question: 10 top https://gateoverflow.in/1964

✔ Answer is 9.

$435 - (110110011)$

num $>>= 1$; implies a num is shifted one bit right in every while loop execution.While loop is executed 9 times successfully and 10th time num is zero.

So count is incremented 9 times.

Note:

Shifting a number "1"bit position to the right will have the effect of dividing by 2:

```
8 >> 1 = $4    // In binary: (00001000) >> 1 = (00000100)
```

36 votes -- Prasanna Ranganathan (3.9k points)

1.11.26 Identify Function: GATE CSE 2014 Set 3 | Question: 10 top https://gateoverflow.in/2044

✔ **A.**

In the computation of given pseudo code for each row and column of Matrix A, each upper triangular element will be interchanged by its mirror image in the lower triangular and after that the same lower triangular element will be again re-interchanged by its mirror image in the upper triangular, resulting the final computed Matrix A same as input Matrix A.

33 votes -- Gate Keeda (15.9k points)

1.11.27 Identify Function: GATE CSE 2015 Set 1 | Question: 31 top https://gateoverflow.in/8263

✔ i loop is executing n times, j loop is executing $\log\ n$ times for each i, and so value of p is $\log n$ when j loop exits for all iterations of the i loop.

k loop is executing $\log\ p$ times, which is $\log\log\ n$ times for each iteration of i. In each of these iteration, q is incremented. So, over all iterations of i, q will be incremented $n\log\log\ n$ times.

So, **D** choice.

77 votes -- Arjun Suresh (330k points)

1.11.28 Identify Function: GATE CSE 2015 Set 2 | Question: 11 top https://gateoverflow.in/8060

✔ $\text{fun}(1) = 1;$

$$\text{fun}(2) = 1 + \text{fun}(1) * \text{fun}(1) = 1 + 1 = 2;$$

$$\text{fun}(3) = 1 + \text{fun}(1) * \text{fun}(2) + \text{fun}(2) * \text{fun}(1) = 5;$$

$$\begin{aligned}\text{fun}(4) &= 1 + \text{fun}(1) * \text{fun}(3) + \text{fun}(2) * \text{fun}(2) + \text{fun}(3) * \text{fun}(1)\\ &= 1 + 5 + 4 + 5 = 15;\end{aligned}$$

$$\text{fun}(5) = 1 + \text{fun}(1) * \text{fun}(4) + \text{fun}(2) * \text{fun}(3) + \text{fun}(3) * \text{fun}(2) + \text{fun}(4) * \text{fun}(1)$$

$= 1 + 15 + 10 + 10 + 15 = 51;$

More formal way:

The recurrence relation is

$$f(n) = \begin{cases} 1, n = 1 \\ 1 + \sum_{i=1}^{n-1} f(i) \times f(n-i), n > 1 \end{cases}$$

$f(1) = 1$

$f(2) = 1 + f(1).\,f(1)$
$\quad = 1 + 1.1 = 2$

$f(3) = 1 + f(1).\,f(2) + f(2).\,f(1)$
$\quad = 1 + 1.2 + 2.1 = 5$

$f(4) = 1 + f(1).\,f(3) + f(2).\,f(2) + f(3).\,f(2)$
$\quad = 1 + 1.5 + 2.2 + 5.1 = 15$

$f(5) = 1 + f(1).\,f(4) + f(2).\,f(3) + f(3).\,f(2) + f(4).\,f(1)$
$\quad = 1 + 1.15 + 2.5 + 5.2 + 15.1 = 51$

109 votes -- Arjun Suresh (330k points)

1.11.29 Identify Function: GATE CSE 2015 Set 3 | Question: 49 top

https://gateoverflow.in/8558

✔ Initially $k = 4$, $c = [1, 0, 1, 1]$, $a = 2$, $n = 8$.

Now let's iterate through the function step by step :

$z = 1$ (at the start of do-something)

$i = 0$ (start of external for loop)

In the do loop

$z = 1 * 1$ (non zero value so considered as true and continue)

$c[0] = 1$, so in the if clause, $z = 1 * 2\%8 = 2$

In the do loop

$z = 2 * 2\%8 = 4$ (since now $z = 2$) (non zero value so considered as true and continue)

$c[0] = 1$, so in the if clause, $z = 4 * 2\%8 = 0$

Now no need to check further :

Reason : All the operations that update Z are multiplicative operations and hence the value of Z will never change from 0.

36 votes -- Tamojit Chatterjee (1.9k points)

1.11.30 Identify Function: GATE CSE 2017 Set 2 | Question: 14 top

https://gateoverflow.in/118245

✔ At first in loop we are giving $x = 0$ then ptr is pointing to X.

So, $*ptr = 0$

Now, we copying the value of ptr to y ,so $Y = 0$

```
x=0;      //value of x = 0
   ptr= &x;      // ptr points to variable x
   y= *ptr;      // Y contain value pointed by ptr i.e. x= 0;
```

Now, value of ptr is changed to 1. so the location of X itself got modified

```
*ptr=1;
```

As it is pointing to x so x will also be changed to 1

So, $1, 0$ will be the value

C is correct answer here.

34 votes -- Aboveallplayer (12.5k points)

1.11.31 Identify Function: GATE CSE 2019 | Question: 18 top https://gateoverflow.in/302830

✔ $x = 2, y = 5$

$y =$ jumble$(5, 2)$ //call by value and y will hold return value. After this call $x = 2,\ y = 12$

$x =$ jumble$(12, 2)$ //call by value and x will hold return value. After this call $x = 26,\ y = 12$

$x = 26$

27 votes -- Digvijay (44.9k points)

1.11.32 Identify Function: GATE CSE 2019 | Question: 26 top https://gateoverflow.in/302822

✔ As per the question, the function convert needs to be called with a **positive integer** n as argument.

So, when a positive int is passed as argument, function will compute $(n/2)$ **which will return the quotient of integer division.**

Again this result of integer division will be passed through compute $(n/2)$ and so on. Each time the no will be divided by 2 and **will gradually become smaller and smaller and close to 0 but it will never be less than 0** .

Hence, the program will not print anything (as after every function call, the value returned is greater than 0) and will not terminate.

PS: Being a C function and hence when run on a computer each recursive call will require creation of activation record consuming memory space. So, eventually the stack memory will run out and program terminates. But such an option is not there. D is the best option here.

Answer (D)

24 votes -- Sayan Bose (5k points)

1.11.33 Identify Function: GATE CSE 2020 | Question: 48 top https://gateoverflow.in/333183

✔ Answer : 81

$pp(3, 4)$

- $a = 3, b = 4, \text{tot} = 1;$

`len = tob(4, array)` will return 3 with array set as 001 as array is updated only once when `b%2 != 0`). The for loop actually iterates 3 times for $b = 4, b = 2$ and $b = 1$, and only when $b = 1$, `arr[i]` is updated.

Now pp will run for loop 3 times:

1. $\text{arr}[0] = 0$. So, $\text{ex} = 3 * 3 = 9$
2. $\text{arr}[1] = 0$. So, $\text{ex} = 9 * 9 = 81$
3. $\text{arr}[1] = 1$. So, $\text{tot} = 1 * 81 = 81$

14 votes -- Prashant Singh (47.1k points)

1.11.34 Identify Function: GATE CSE 2021 Set 1 | Question: 48 top https://gateoverflow.in/357403

✔ For index i, we get $\text{total}_i = x * \text{total}_{i-1} + Y_i$. We have $Y_i = 1, x = 2, \text{total}_0 = 1$.

So, $\text{total}_n = 2 * \text{total}_{n-1} + 1$

$$= 2 * (2 * \text{total}_{n-2} + 1) + 1$$

$$= 2^2 * \text{total}_{n-2} + 2 + 1$$

$$\vdots$$

$$= 2^2 * \text{total}_{n-2} + 2 + 1$$

$$= 2^n * \text{total}_{n-n} + (2^{n-1} + 2^{n-2} + \ldots + 2 + 1)$$

$$= 2^n * \text{total}_0 + (2^n - 1)$$

$$= 2^{n+1} - 1$$

So, $\text{total}_9 = 2^{10} - 1 = 1023$.

4 votes -- gatecse (62.6k points)

1.11.35 Identify Function: GATE CSE 2021 Set 2 | Question: 23 top https://gateoverflow.in/357517

✔ This function is calculating the GCD of the two numbers by repeated subtraction.

$\text{GCD}(15, 255) = 15$. So it'll return 15.

3 votes -- zxy123 (2.5k points)

1.11.36 Identify Function: GATE IT 2005 | Question: 53 top https://gateoverflow.in/3814

✔ The answer is (**D**).

```
#include <stdio.h>

int main(void) {

return 0;
}

int anagram (char *a, char *b) {
/*
ASCII characters are of 7-bits
so we use count array to represent all the ASCII characters
(ranging 0-127)
*/
int count [128], j;

/*
so this loop will initialize count of all the ASCII characters to be
0 (zero)
*/
for (j = 0;  j < 128; j++) count[j] = 0;

j = 0;
/*
"a[j] && b[j]"  ensures that anagram returns 0 (false) in case both
strings have different length. Because different length strings cannot
be anagram of each other
*/

/*
Logic:

Below while loop increments ASCII equivalent position for its occurence
in array 'a'in count array; and decrements ASCII equivalent position
for its occurence in array 'b'in count array.

Example: a = "ISS" and b = "SIS"
ASCII equivalent of:
I - 73
```

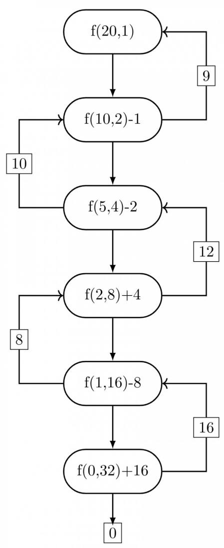
```
S - 83

j = 0: Statement A will increment count[ASCII of 'I']==>count[73]
count[73] = 0 --> 1
Statement B will decrement count[ASCII of 'S'] ==> count[83]
count[83] = 0 --> -1 and will increment j j = 0 --> 1

j = 1: Statement A will increment count[ASCII of 'S'] ==> count[83]
count[83] = -1 --> 0
Statement B will decrement count[ASCII of 'I'] ==> count[73]
count[73] = 1 --> 0 and will increment j j = 1 --> 2

j = 2:  Statement A will increment count[ASCII of 'S'] ==> count[83]
count[83] = 0 --> 1
Statement B will decrement count[ASCII of 'S'] ==> count[83]
count[83] = 1 --> 0 and will increment j j = 2 --> 3

*** END OF LOOP ***

*/
while (a[j] && b[j]) {


A; //count [a[j]]++

/*
Note: j will be increment after count[]-- will execute
Resource: http://www.c4learn.com/c-programming/increment-operator-inside-printf
*/
B;  //count[b[j++]]--
}

/*
This loop checks that the number of occurences of the individual ASCII
characters is same or not.
If count[i] = 0 ---> same number of occurences for ASCII chracter i
---> return 1 (true)

if count[i]!= 0 ---> different number of occurences for ASCII chracter i
---> return 0 (false)
*/

for (j = 0; j < 128; j++) if (count [j]) return 0;
return 1;
}
```

54 votes -- Sohil Ladhani (137 points)

1.11.37 Identify Function: GATE IT 2005 | Question: 57 top

☛ https://gateoverflow.in/3818

✔ See the following calling sequence. Boxed values show the return values.



Hence, answer is option **C**.

26 votes -- Rajesh Pradhan (18.9k points)

1.11.38 Identify Function: GATE IT 2006 | Question: 52 top https://gateoverflow.in/3595

✔ Answer: **C**

Because $\binom{m}{0} = 1$ and $\binom{n}{n} = 1$.

28 votes -- Rajarshi Sarkar (27.8k points)

1.11.39 Identify Function: GATE IT 2008 | Question: 82 top https://gateoverflow.in/3406

✔ Answer: **D**

The function prints the binary equivalent of the number n.

Binary equivalent of 173 is 10101101.

20 votes -- Rajarshi Sarkar (27.8k points)

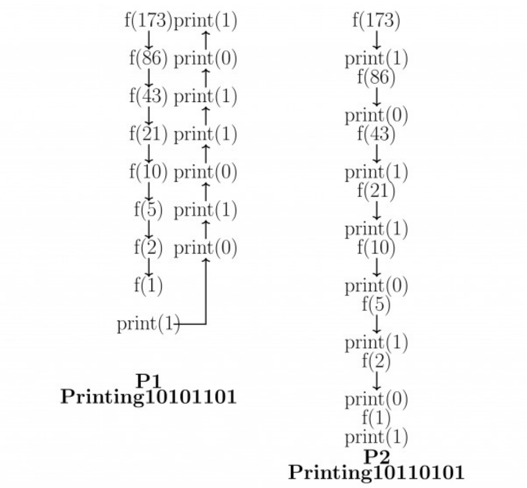
1.11.40 Identify Function: GATE IT 2008 | Question: 83 top https://gateoverflow.in/3407

✔ Answer: C

The code fragment written in C and P1 prints the binary equivalent of the number n.

P2 prints the binary equivalent of the number n in reverse.

22 votes -- Rajarshi Sarkar (27.8k points)



Here, $P1$ and $P2$ will print opposite in direction as shown in diagram.
And given code fragment will print like $P1$ and not like $P2$
Hence, answer will be (**C**).

27 votes -- srestha (85.2k points)

1.12 Minimum Maximum (1) top

1.12.1 Minimum Maximum: GATE CSE 2014 Set 1 | Question: 39 top https://gateoverflow.in/1917

The minimum number of comparisons required to find the minimum and the maximum of 100 numbers is ________

gate2014-cse-set1 algorithms numerical-answers normal minimum-maximum

Answer

Answers: Minimum Maximum

1.12.1 Minimum Maximum: GATE CSE 2014 Set 1 | Question: 39 top https://gateoverflow.in/1917

✔ We can solve this question by using **Tournament Method Technique** -

1. To find the smallest element in the array will take $n-1$ comparisons $= 99$.
2. To find the largest element -

a. After the first round of Tournament , there will be exactly $n/2$ numbers $= 50$ that will loose the round.
b. So, the biggest looser (the largest number) should be among these 50 loosers.To find the largest number will take $n/2-1$ comparisons $= 49$.

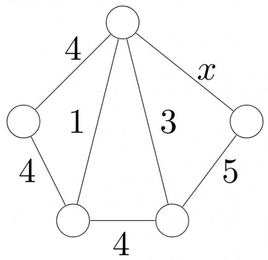
Total $99 + 49 = 148$.

94 votes -- Harsh181996 (3k points)

1.13 Minimum Spanning Trees (4) top

1.13.1 Minimum Spanning Trees: GATE CSE 2018 | Question: 47 top https://gateoverflow.in/204122

Consider the following undirected graph G:



Choose a value for x that will maximize the number of minimum weight spanning trees (MWSTs) of G. The number of MWSTs of G for this value of x is ____.

gate2018-cse algorithms graph-algorithms minimum-spanning-trees numerical-answers

Answer

1.13.2 Minimum Spanning Trees: GATE CSE 2020 | Question: 31 top https://gateoverflow.in/333200

Let $G=(V,E)$ be a weighted undirected graph and let T be a Minimum Spanning Tree (MST) of G maintained using adjacency lists. Suppose a new weighed edge $(u,v) \in V \times V$ is added to G. The worst case time complexity of determining if T is still an MST of the resultant graph is

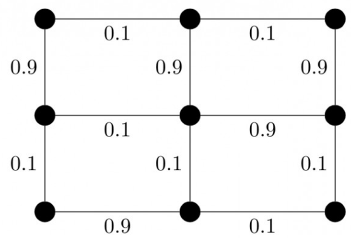
A. $\Theta(\mid E \mid + \mid V \mid)$
B. $\Theta(\mid E \mid \mid V \mid)$
C. $\Theta(E \mid \log \mid V \mid)$
D. $\Theta(\mid V \mid)$

gate2020-cse algorithms minimum-spanning-trees graph-algorithms

Answer

1.13.3 Minimum Spanning Trees: GATE CSE 2021 Set 1 | Question: 17 top https://gateoverflow.in/357434

Consider the following undirected graph with edge weights as shown:



The number of minimum-weight spanning trees of the graph is ___________.

gate2021-cse-set1 algorithms graph-algorithms minimum-spanning-trees numerical-answers

Answer

1.13.4 Minimum Spanning Trees: GATE CSE 2021 Set 2 | Question: 1 top https://gateoverflow.in/357539

Let G be a connected undirected weighted graph. Consider the following two statements.

- S_1: There exists a minimum weight edge in G which is present in every minimum spanning tree of G.
- S_2: If every edge in G has distinct weight, then G has a unique minimum spanning tree.

Which one of the following options is correct?

A. Both S_1 and S_2 are true
B. S_1 is true and S_2 is false
C. S_1 is false and S_2 is true
D. Both S_1 and S_2 are false

gate2021-cse-set2 algorithms graph-algorithms minimum-spanning-trees

Answer

Answers: Minimum Spanning Trees

1.13.1 Minimum Spanning Trees: GATE CSE 2018 | Question: 47 top https://gateoverflow.in/204122

✔ Number of possible MSTs increase, when we have multiple edges with same edge weights.

To maximize the number of MST, x should be 5.
In the question, number of MST is asked for the value of X.

So, number of MST $= 2 \times 2 = 4$ **(Answer)**
(Because one 4 forms cycle, cant be included in any way. Now from two 4 and 5 we can select one in $2 \times 2 = 4$ ways)

33 votes -- Ahwan Mishra (10.2k points)

1.13.2 Minimum Spanning Trees: GATE CSE 2020 | Question: 31 top https://gateoverflow.in/333200

✔ We can do this in $O(|V|)$ in the following way:

1. Run BFS in T from u to v to detect the edge with maximum value in that path. $-O(|V|)$.
2. If the weight of that edge is greater than the weight of the edge you're trying to add, remove that old edge and add the new one.
3. Otherwise, do nothing, because the new edge would not improve the MST. $-O(1)$.

The rationale behind this solution is that simply adding an edge into T would create exactly one cycle, and to restore the MST property we have to remove the edge with maximum value from that cycle.

Source: https://stackoverflow.com/questions/30881340/update-minimum-spanning-tree-if-edge-is-added

References

22 votes -- goxul (5.2k points)

1.13.3 Minimum Spanning Trees: GATE CSE 2021 Set 1 | Question: 17 top https://gateoverflow.in/357434

✔ For finding a minimum-weight spanning tree we have different algorithms like the Prim, Kruskal's algorithm.

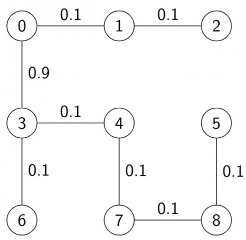
A spanning tree means a sub-graph of the original graph that should be the tree and connected to all the vertices together.

A Minimum spanning tree is for a weighted, connected and undirected graph is a spanning tree with weight less than or equal to the weight of every other spanning tree. The weight of a spanning tree is the sum of the weights of each edge of the spanning tree.

Now to find out MST of the given graph using Kruskal's algorithm (we can use any method here) we can follow the 2 step process:

1. Sort all the edges in increasing order of their weight.
2. Select the smallest edge & check if it forms a cycle with the spanning tree formed so far. If a cycle is not formed, include this edge. Else, discard it.

After applying the above steps graph will look like this:

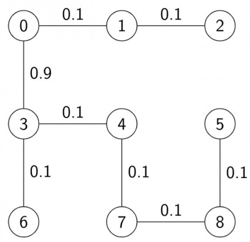


In the above figure, we cannot add edges $(6,7)$ or $(4,5)$ because it creates a cycle in MST. So we have only 3 edges remaining which are $(0,3)$, $(1,4)$ and $(2,5)$.

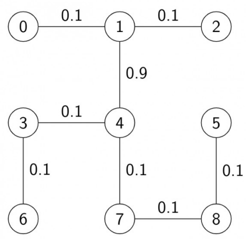
Since we know that for n vertices, the number of edges should be $(n-1)$ in MST. so from 3 edges we can select only 1 edge that is :

$\binom{3}{1} = 3$ ways.

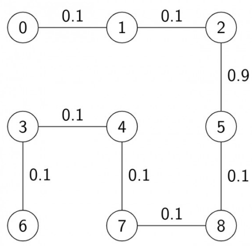
- $1^{st}\,MST$



- $2^{nd}\,MST$:



- $3^{rd}\,MST$:



$\therefore$ Total 3 MSTs are possible for the given graph.

For more Refer here:[Kruskal Algorithm]

References

👍 7 votes -- Hira (13.8k points)

1.13.4 Minimum Spanning Trees: GATE CSE 2021 Set 2 | Question: 1 top

☛ https://gateoverflow.in/357539

✔ We can think of running the Kruskals algorithm for finding the Minimum Spanning Tree on graph G.

While doing that, we sort the edges based on their weight and start selecting edges from the smallest weight (w_{small} for example).

Problem with S_1: If we have multiple copies of w_{small}, then a specific w_{small} weighted edge is not guaranteed to be selected by Kruskals algorithm.

S_2 is Correct: If the sorted order of the edges contains only distinct values, Kruskals algorithm will always select a unique set of edges resulting in a unique minimum spanning tree.

Correct option: C.

1 votes -- Debashish Deka (40.7k points)

1.14 Quicksort (5) top

1.14.1 Quicksort: GATE CSE 1987 | Question: 1-xviii top https://gateoverflow.in/80366

Let P be a quicksort program to sort numbers in ascending order. Let t_1 and t_2 be the time taken by the program for the inputs $[1\ 2\ 3\ 4]$ and $[5\ 4\ 3\ 2\ 1]$, respectively. Which of the following holds?

A. $t_1 = t_2$
B. $t_1 > t_2$
C. $t_1 < t_2$
D. $t_1 = t_2 + 5 \log 5$

gate1987 algorithms sorting quicksort

Answer

1.14.2 Quicksort: GATE CSE 1989 | Question: 9 top https://gateoverflow.in/89083

An input files has 10 records with keys as given below:

$25 \quad 7 \quad 34 \quad 2 \quad 70 \quad 9 \quad 61 \quad 16 \quad 49 \quad 19$

This is to be sorted in non-decreasing order.

i. Sort the input file using QUICKSORT by correctly positioning the first element of the file/subfile. Show the subfiles obtained at all intermediate steps. Use square brackets to demarcate subfiles.
ii. Sort the input file using 2-way- MERGESORT showing all major intermediate steps. Use square brackets to demarcate subfiles.

gate1989 descriptive algorithms sorting quicksort

Answer

1.14.3 Quicksort: GATE CSE 1992 | Question: 03,iv top https://gateoverflow.in/581

Assume that the last element of the set is used as partition element in Quicksort. If n distinct elements from the set $[1 \dots n]$ are to be sorted, give an input for which Quicksort takes maximum time.

gate1992 algorithms sorting easy quicksort descriptive

Answer

1.14.4 Quicksort: GATE CSE 1994 | Question: 1.19, ISRO2016-31 top https://gateoverflow.in/2462

Algorithm design technique used in quicksort algorithm is?

A. Dynamic programming
B. Backtracking
C. Divide and conquer
D. Greedy method

gate1994 algorithms sorting quicksort easy isro2016

Answer

1.14.5 Quicksort: GATE CSE 2019 | Question: 20 top https://gateoverflow.in/302828

An array of 25 distinct elements is to be sorted using quicksort. Assume that the pivot element is chosen uniformly at random. The probability that the pivot element gets placed in the worst possible location in the first round of partitioning (rounded off to 2 decimal places) is ________

gate2019-cse numerical-answers algorithms quicksort probability

Answer

Answers: Quicksort

1.14.1 Quicksort: GATE CSE 1987 | Question: 1-xviii top https://gateoverflow.in/80366

✔ Actually, in both the cases, it will take $O(n^2)$ time for partition algorithm and $T(n-1)$ time for subproblem. As n is the number of inputs and in the 2^{nd} case inputs are 5(greater than 1^{st} one that is 4), $t_1 < t_2$.

Correct Answer: C.

42 votes -- Rohan Ghosh (1.6k points)

1.14.2 Quicksort: GATE CSE 1989 | Question: 9 top https://gateoverflow.in/89083

✔ **Quick Sort**

Pseudocode for quicksort

QUICKSORT(A,p,r)
  if $p < r$
    then $q \leftarrow \text{PARTITION}(A, p, r)$
      $\text{QUICKSORT}(A, p, q-1)$
      $\text{QUICKSORT}(A, q+1, r)$

Initial call: $\text{QUICKSORT}(A, 1, n)$

1. 25 7 34 2 70 9 61 16 49 19
2. 25 7 2 34 70 9 61 16 49 19
3. 25 7 2 9 70 34 61 16 49 19
4. 25 7 2 9 16 34 61 70 49 19
5. 25 7 2 9 16 19 61 70 49 34
6. [7 2 9 16 19] 25 [61 70 49 34]
7. [7 2 9 16 19] 25 [61 70 49 34]
8. [[2] 7 [9 16 19]] 25 [61 70 49 34]
9. [2 7 [9 [16 19]]] 25 [61 70 49 34]
10. [2 7 [9 [16 19]]] 25 [61 70 49 34]
11. [2 7 [9 [16 19]]] 25 [61 70 49 34]
12. [2 7[9 16 19]] 25 [61 70 49 34]
13. [2 7 9 16 19] 25 [61 70 49 34]
14. [2 7 9 16 19] 25 [61 49 70 34]
15. [2 7 9 16 19] 25 [61 49 34 70]
16. [2 7 9 16 19] 25 [[49 34] 61 [70]]
17. [2 7 9 16 19] 25 [[49 34] 61 [70]]
18. [2 7 9 16 19] 25 [[[34] 49] 61 [70]]
19. [2 7 9 16 19] 25 [[[34] 49] 61 [70]]
20. [2 7 9 16 19] 25 [[34 49] 61 [70]]
21. [2 7 9 16 19] 25 [34 49 61 [70]]
22. [2 7 9 16 19] 25 [34 49 61 70]
23. [2 7 9 16 19 25 34 49 61 70]

2-way MergeSort [Iterative version]

1. [25 7] [34 2] [70 9] [61 16] [49 19]
2. [[7 25] [2 34]] [[9 70] [16 61]] [[19 49]]
3. [[2 7 25 34] [9 16 61 70]] [19 49]
4. [[2 7 9 16 25 34 61 70] [19 49]]
5. [[2 7 9 16 19 25 34 49 61 70]

MergeSort [Recursive version]
MERGE - SORT A[1...n]

1. If $n = 1$, done.
2. Recursively sort $A[1 \ldots \lceil n/2 \rceil]$ and $A[\lceil n/2 \rceil + 1 \ldots n]$.
3. "Merge" the 2 sorted lists.

Key subroutine: **MERGE**

1. [25 7 34 2 70] [9 61 16 49 19]
2. [[25 7 34] 2 70] [9 61 16 49 19]
3. [[[7 25] [34]] [2 70]] [9 61 16 49 19]
4. [[[7 25] [34]] [2 70]] [9 61 16 49 19]
5. [[[7 25 34]] [2 70]] [9 61 16 49 19]
6. [[7 25 34] [2 70]] [9 61 16 49 19]
7. [2 7 25 34 70] [9 61 16 49 19]
8. [2 7 25 34 70] [[9 61 16] [49 19]]
9. [2 7 25 34 70] [[[9 61] [16]] [49 19]]
10. [2 7 25 34 70] [[[9 61] [16]] [49 19]]
11. [2 7 25 34 70] [[[9 61] [16]] [49 19]]
12. [2 7 25 34 70] [[9 16 61] [49 19]]
13. [2 7 25 34 70] [[9 16 61] [19 49]]
14. [2 7 25 34 70] [9 16 19 49 61]
15. [2 7 9 16 19 25 34 49 61 70]

12 votes -- Satbir Singh (21k points)

1.14.3 Quicksort: GATE CSE 1992 | Question: 03,iv top https://gateoverflow.in/581

✔ The algorithm will take maximum time when:

1. The array is already sorted in same order.(Here,ascending order)
2. The array is already sorted in reverse order.
3. All elements are same in the array(Here,we have n distinct elements).So, we can say the above points (1) and (2) as answers.

38 votes -- Rajarshi Sarkar (27.8k points)

1.14.4 Quicksort: GATE CSE 1994 | Question: 1.19, ISRO2016-31 top https://gateoverflow.in/2462

✔ **Answer: Option C.**

It is one of the efficient algorithms in **Divide and Conquer** strategy.

25 votes -- Gate Keeda (15.9k points)

1.14.5 Quicksort: GATE CSE 2019 | Question: 20 top https://gateoverflow.in/302828

✔ Worst case of quicksort, if pivot element is Minimum or Maximum.

Total elements $= 25$
For worst case number of candidates $= 2$

$P = \frac{2}{25} = 0.08$

34 votes -- Digvijay (44.9k points)

1.15 Recurrence (33) top

1.15.1 Recurrence: GATE CSE 1987 | Question: 10a top https://gateoverflow.in/82450

Solve the recurrence equations:

- $T(n) = T(n-1) + n$
- $T(1) = 1$

gatel987 algorithms recurrence descriptive

Answer

1.15.2 Recurrence: GATE CSE 1988 | Question: 13iv top

https://gateoverflow.in/94637

Solve the recurrence equations:

- $T(n) = T(\frac{n}{2}) + 1$
- $T(1) = 1$

gate1988 descriptive algorithms recurrence

Answer

1.15.3 Recurrence: GATE CSE 1989 | Question: 13b top

https://gateoverflow.in/93175

Find a solution to the following recurrence equation:

- $T(n) = \sqrt{n} + T\left(\frac{n}{2}\right)$
- $T(1) = 1$

gate1989 descriptive algorithms recurrence

Answer

1.15.4 Recurrence: GATE CSE 1990 | Question: 17a top

https://gateoverflow.in/86878

Express $T(n)$ in terms of the harmonic number $H_n = \sum_{i=1}^{n} \frac{1}{i}, \quad n \geq 1$, where $T(n)$ satisfies the recurrence relation,

$T(n) = \frac{n+1}{n} T(n-1) + 1$, for $n \geq \sum$ and $T(1) = 1$

What is the asymptotic behaviour of $T(n)$ as a function of n ?

gate1990 descriptive algorithms recurrence

Answer

1.15.5 Recurrence: GATE CSE 1992 | Question: 07a top

https://gateoverflow.in/586

Consider the function $F(n)$ for which the pseudocode is given below :

```
Function F(n)
begin
F1 ← 1
if(n=1) then F ← 3
 else
   For i = 1 to n do
        begin
            C ← 0
           For j = 1 to n - 1 do
           begin C ← C + 1 end
           F1 = F1 * C
       end
 F = F1
end
```

[n is a positive integer greater than zero]

A. Derive a recurrence relation for $F(n)$.

gate1992 algorithms recurrence descriptive

Answer

1.15.6 Recurrence: GATE CSE 1992 | Question: 07b top https://gateoverflow.in/43600

Consider the function $F(n)$ for which the pseudocode is given below :

```
Function F(n)
begin
F1 ← 1
if(n=1) then F ← 3
 else
   For i = 1 to n do
        begin
            C ← 0
           For j = 1 to n – 1 do
           begin C ← C + 1 end
           F1 = F1 * C
       end
 F = F1
end
```

[n is a positive integer greater than zero]

B. Solve the recurrence relation for a closed form solution of $F(n)$.

gate1992 algorithms recurrence descriptive

Answer

1.15.7 Recurrence: GATE CSE 1993 | Question: 15 top https://gateoverflow.in/2312

Consider the recursive algorithm given below:

```
procedure bubblesort (n);
var i,j: index; temp : item;
begin
    for i:=1 to n-1 do
        if A[i] > A[i+1] then
        begin
            temp := A[i];
            A[i] := A[i+1];
            A[i+1] := temp;
        end;
    bubblesort (n-1)
end
```

Let a_n be the number of times the 'if…then…' statement gets executed when the algorithm is run with value n. Set up the recurrence relation by defining a_n in terms of a_{n-1}. Solve for a_n.

gate1993 algorithms recurrence normal descriptive

Answer

1.15.8 Recurrence: GATE CSE 1994 | Question: 1.7, ISRO2017-14 top https://gateoverflow.in/2444

The recurrence relation that arises in relation with the complexity of binary search is:

A. $T(n) = 2T\left(\frac{n}{2}\right) + k$, k is a constant
B. $T(n) = T\left(\frac{n}{2}\right) + k$, k is a constant
C. $T(n) = T\left(\frac{n}{2}\right) + \log n$
D. $T(n) = T\left(\frac{n}{2}\right) + n$

gate1994 algorithms recurrence easy isro2017

Answer

1.15.9 Recurrence: GATE CSE 1996 | Question: 2.12 top https://gateoverflow.in/2741

The recurrence relation

- $T(1) = 2$
- $T(n) = 3T(\frac{n}{4}) + n$

has the solution $T(n)$ equal to

A. $O(n)$
B. $O(\log n)$
C. $O\left(n^{\frac{3}{4}}\right)$
D. None of the above

gate1996 algorithms recurrence normal

Answer

1.15.10 Recurrence: GATE CSE 1997 | Question: 15 top https://gateoverflow.in/2275

Consider the following function.

```
Function F(n, m:integer):integer;
begin
    if (n<=0) or (m<=0) then F:=1
    else
      F:F(n-1, m) + F(n-1, m-1);
end;
```

Use the recurrence relation $\begin{pmatrix} n \\ k \end{pmatrix} = \begin{pmatrix} n-1 \\ k \end{pmatrix} + \begin{pmatrix} n-1 \\ k-1 \end{pmatrix}$ to answer the following questions. Assume that n, m are positive integers. Write only the answers without any explanation.

a. What is the value of $F(n, 2)$?
b. What is the value of $F(n, m)$?
c. How many recursive calls are made to the function F, including the original call, when evaluating $F(n, m)$.

gate1997 algorithms recurrence descriptive

Answer

1.15.11 Recurrence: GATE CSE 1997 | Question: 4.6 top https://gateoverflow.in/2247

Let $T(n)$ be the function defined by $T(1) = 1,\ T(n) = 2T(\lfloor \frac{n}{2} \rfloor) + \sqrt{n}$ for $n \geq 2$.

Which of the following statements is true?

A. $T(n) = O\sqrt{n}$
B. $T(n) = O(n)$
C. $T(n) = O(\log n)$
D. None of the above

gate1997 algorithms recurrence normal

Answer

1.15.12 Recurrence: GATE CSE 1998 | Question: 6a top https://gateoverflow.in/44584

Solve the following recurrence relation

$x_n = 2x_{n-1} - 1, n > 1$

$x_1 = 2$

gate1998 algorithms recurrence descriptive

Answer

1.15.13 Recurrence: GATE CSE 2002 | Question: 1.3 top https://gateoverflow.in/807

The solution to the recurrence equation $T(2^k) = 3T(2^{k-1}) + 1, T(1) = 1$ is

A. 2^k
B. $\frac{(3^{k+1}-1)}{2}$

C. $3^{\log_2 k}$
D. $2^{\log_3 k}$

gate2002-cse algorithms recurrence normal

Answer

1.15.14 Recurrence: GATE CSE 2002 | Question: 2.11 top

☛ https://gateoverflow.in/841

The running time of the following algorithm

Procedure $A(n)$

If $n \leqslant 2$ return (1) else return $(A(\lceil\sqrt{n}\rceil))$;

is best described by

A. $O(n)$
B. $O(\log n)$
C. $O(\log\log n)$
D. $O(1)$

gate2002-cse algorithms recurrence normal

Answer

1.15.15 Recurrence: GATE CSE 2003 | Question: 35 top

☛ https://gateoverflow.in/925

Consider the following recurrence relation

$T(1) = 1$

$T(n+1) = T(n) + \lfloor\sqrt{n+1}\rfloor$ for all $n \geq 1$

The value of $T(m^2)$ for $m \geq 1$ is

A. $\frac{m}{6}(21m - 39) + 4$
B. $\frac{m}{6}\left(4m^2 - 3m + 5\right)$
C. $\frac{m}{2}\left(3m^{2.5} - 11m + 20\right) - 5$
D. $\frac{m}{6}\left(5m^3 - 34m^2 + 137m - 104\right) + \frac{5}{6}$

gate2003-cse algorithms time-complexity recurrence difficult

Answer

1.15.16 Recurrence: GATE CSE 2004 | Question: 83, ISRO2015-40 top

☛ https://gateoverflow.in/1077

The time complexity of the following C function is (assume $n > 0$)

```
int recursive (int n) {
    if(n == 1)
        return (1);
    else
        return (recursive (n-1) + recursive (n-1));
}
```

A. $O(n)$
B. $O(n\log n)$
C. $O(n^2)$
D. $O(2^n)$

gate2004-cse algorithms recurrence time-complexity normal isro2015

Answer

1.15.17 Recurrence: GATE CSE 2004 | Question: 84 top

☛ https://gateoverflow.in/1078

The recurrence equation

$T(1) = 1$

$T(n) = 2T(n-1) + n, n \geq 2$

evaluates to

A. $2^{n+1} - n - 2$
B. $2^n - n$
C. $2^{n+1} - 2n - 2$
D. $2^n + n$

gate2004-cse algorithms recurrence normal

Answer

1.15.18 Recurrence: GATE CSE 2006 | Question: 51, ISRO2016-34 top

https://gateoverflow.in/1829

Consider the following recurrence:

$$T(n) = 2T\left(\sqrt{n}\right) + 1,\ T(1) = 1$$

Which one of the following is true?

A. $T(n) = \Theta(\log\log n)$
B. $T(n) = \Theta(\log n)$
C. $T(n) = \Theta(\sqrt{n})$
D. $T(n) = \Theta(n)$

algorithms recurrence isro2016 gate2006-cse

Answer

1.15.19 Recurrence: GATE CSE 2008 | Question: 78 top

https://gateoverflow.in/497

Let x_n denote the number of binary strings of length n that contain no consecutive 0s.

Which of the following recurrences does x_n satisfy?

A. $x_n = 2x_{n-1}$
B. $x_n = x_{\lfloor n/2 \rfloor} + 1$
C. $x_n = x_{\lfloor n/2 \rfloor} + n$
D. $x_n = x_{n-1} + x_{n-2}$

gate2008-cse algorithms recurrence normal

Answer

1.15.20 Recurrence: GATE CSE 2008 | Question: 79 top

https://gateoverflow.in/43485

Let x_n denote the number of binary strings of length n that contain no consecutive 0s.

The value of x_5 is

A. 5
B. 7
C. 8
D. 16

gate2008-cse algorithms recurrence normal

Answer

1.15.21 Recurrence: GATE CSE 2009 | Question: 35 top

https://gateoverflow.in/1321

The running time of an algorithm is represented by the following recurrence relation:

$$T(n) = \begin{cases} n & n \leq 3 \\ T(\frac{n}{3}) + cn & \text{otherwise} \end{cases}$$

Which one of the following represents the time complexity of the algorithm?

A. $\Theta(n)$

B. $\Theta(n\log n)$

C. $\Theta(n^2)$
D. $\Theta(n^2 \log n)$

gate2009-cse algorithms recurrence time-complexity normal

Answer

1.15.22 Recurrence: GATE CSE 2012 | Question: 16 top https://gateoverflow.in/48

The recurrence relation capturing the optimal execution time of the $Towers\ of\ Hanoi$ problem with n discs is

A. $T(n) = 2T(n-2) + 2$
B. $T(n) = 2T(n-1) + n$
C. $T(n) = 2T(n/2) + 1$
D. $T(n) = 2T(n-1) + 1$

gate2012-cse algorithms easy recurrence

Answer

1.15.23 Recurrence: GATE CSE 2014 Set 2 | Question: 13 top https://gateoverflow.in/1968

Which one of the following correctly determines the solution of the recurrence relation with $T(1) = 1$?

$$T(n) = 2T\left(\frac{n}{2}\right) + \log n$$

A. $\Theta(n)$
B. $\Theta(n \log n)$
C. $\Theta(n^2)$
D. $\Theta(\log n)$

gate2014-cse-set2 algorithms recurrence normal

Answer

1.15.24 Recurrence: GATE CSE 2015 Set 1 | Question: 49 top https://gateoverflow.in/8355

Let a_n represent the number of bit strings of length n containing two consecutive 1s. What is the recurrence relation for a_n?

A. $a_{n-2} + a_{n-1} + 2^{n-2}$
B. $a_{n-2} + 2a_{n-1} + 2^{n-2}$
C. $2a_{n-2} + a_{n-1} + 2^{n-2}$
D. $2a_{n-2} + 2a_{n-1} + 2^{n-2}$

gate2015-cse-set1 algorithms recurrence normal

Answer

1.15.25 Recurrence: GATE CSE 2015 Set 3 | Question: 39 top https://gateoverflow.in/8498

Consider the following recursive C function.

```
void get(int n)
{
    if (n<1) return;
    get (n-1);
    get (n-3);
    printf("%d", n);
}
```

If $get(6)$ function is being called in $main()$ then how many times will the $get()$ function be invoked before returning to the $main()$?

A. 15
B. 25
C. 35
D. 45

gate2015-cse-set3 algorithms recurrence normal

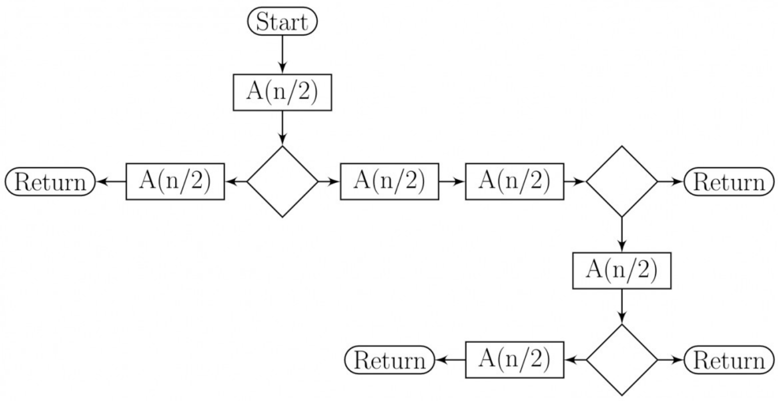
Answer

1.15.26 Recurrence: GATE CSE 2016 Set 2 | Question: 39 top

https://gateoverflow.in/39581

The given diagram shows the flowchart for a recursive function $A(n)$. Assume that all statements, except for the recursive calls, have $O(1)$ time complexity. If the worst case time complexity of this function is $O(n^{\alpha})$, then the least possible value (accurate up to two decimal positions) of α is ________.

Flow chart for Recursive Function $A(n)$.



gate2016-cse-set2 algorithms time-complexity recurrence normal numerical-answers

Answer

1.15.27 Recurrence: GATE CSE 2017 Set 2 | Question: 30 top

https://gateoverflow.in/118623

Consider the recurrence function

$$T(n) = \begin{cases} 2T(\sqrt{n}) + 1, & n > 2 \\ 2, & 0 < n \leq 2 \end{cases}$$

Then $T(n)$ in terms of Θ notation is

A. $\Theta(\log \log n)$
B. $\Theta(\log n)$
C. $\Theta(\sqrt{n})$
D. $\Theta(n)$

gate2017-cse-set2 algorithms recurrence

Answer

1.15.28 Recurrence: GATE CSE 2020 | Question: 2 top

https://gateoverflow.in/333229

For parameters a and b, both of which are $\omega(1)$, $T(n) = T(n^{1/a}) + 1$, and $T(b) = 1$. Then $T(n)$ is

A. $\Theta(\log_a \log_b n)$
B. $\Theta(\log_{ab} n)$
C. $\Theta(\log_b \log_a n)$
D. $\Theta(\log_2 \log_2 n)$

gate2020-cse algorithms recurrence

Answer

1.15.29 Recurrence: GATE CSE 2021 Set 1 | Question: 30 top

https://gateoverflow.in/357421

Consider the following recurrence relation.

$$T(n) = \begin{cases} T(n/2) + T(2n/5) + 7n & \text{if } n > 0 \\ 1 & \text{if } n = 0 \end{cases}$$

Which one of the following options is correct?

A. $T(n) = \Theta(n^{5/2})$
B. $T(n) = \Theta(n \log n)$
C. $T(n) = \Theta(n)$
D. $T(n) = \Theta((\log n)^{5/2})$

gate2021-cse-set1 algorithms recurrence time-complexity

Answer

1.15.30 Recurrence: GATE CSE 2021 Set 2 | Question: 39 top

https://gateoverflow.in/357501

For constants $a \geq 1$ and $b > 1$, consider the following recurrence defined on the non-negative integers:

$$T(n) = a \cdot T\left(\frac{n}{b}\right) + f(n)$$

Which one of the following options is correct about the recurrence $T(n)$?

A. If $f(n)$ is $n \log_2(n)$, then $T(n)$ is $\Theta(n \log_2(n))$
B. If $f(n)$ is $\dfrac{n}{\log_2(n)}$, then $T(n)$ is $\Theta(\log_2(n))$
C. If $f(n)$ is $O(n^{\log_b(a)-\epsilon})$ for some $\epsilon > 0$, then $T(n)$ is $\Theta(n^{\log_b(a)})$
D. If $f(n)$ is $\Theta(n^{\log_b(a)})$, then $T(n)$ is $\Theta(n^{\log_b(a)})$

gate2021-cse-set2 algorithms recurrence

Answer

1.15.31 Recurrence: GATE IT 2004 | Question: 57 top

https://gateoverflow.in/3700

Consider a list of recursive algorithms and a list of recurrence relations as shown below. Each recurrence relation corresponds to exactly one algorithm and is used to derive the time complexity of the algorithm.

	Recursive Algorithm		Recurrence Relation
P	Binary search	I.	$T(n) = T(n-k) + T(k) + cn$
Q.	Merge sort	II.	$T(n) = 2T(n-1) + 1$
R.	Quick sort	III.	$T(n) = 2T(n/2) + cn$
S.	Tower of Hanoi	IV.	$T(n) = T(n/2) + 1$

Which of the following is the correct match between the algorithms and their recurrence relations?

A. P-II, Q-III, R-IV, S-I
B. P-IV, Q-III, R-I, S-II
C. P-III, Q-II, R-IV, S-I
D. P-IV, Q-II, R-I, S-III

gate2004-it algorithms recurrence normal

Answer

1.15.32 Recurrence: GATE IT 2005 | Question: 51 top

https://gateoverflow.in/3812

Let $T(n)$ be a function defined by the recurrence

$T(n) = 2T(n/2) + \sqrt{n}$ for $n \geq 2$ and
$T(1) = 1$

Which of the following statements is **TRUE**?

A. $T(n) = \Theta(\log n)$

B. $T(n) = \Theta(\sqrt{n})$
C. $T(n) = \Theta(n)$
D. $T(n) = \Theta(n \log n)$

gate2005-it algorithms recurrence easy

Answer

1.15.33 Recurrence: GATE IT 2008 | Question: 44 top

https://gateoverflow.in/3354

When $n = 2^{2k}$ for some $k \geqslant 0$, the recurrence relation

$T(n) = \sqrt(2)T(n/2) + \sqrt{n}, T(1) = 1$

evaluates to :

A. $\sqrt(n)(\log n + 1)$
B. $\sqrt(n) \log n$
C. $\sqrt(n) \log \sqrt(n)$
D. $n \log \sqrt{n}$

gate2008-it algorithms recurrence normal

Answer

Answers: Recurrence

1.15.1 Recurrence: GATE CSE 1987 | Question: 10a top

https://gateoverflow.in/82450

✔ $$\begin{aligned} T(n) &= T(n-1) + n \\ &= T(n-2) + (n-1) + n \\ &= T(n-3) + (n-2) + (n-1) + n \\ &. \\ &. \\ &. \\ &= T(n-k) + [(n-k+1) + (n-k+2) + \ldots + (n-1) + n] \end{aligned}$$

Recurrence stops at,

$n - k = 1$
$k = n - 1$

$\therefore T(n) = T(1) + [2 + 3 + \ldots + n] = 1 + 2 + 3 + \ldots + n = \frac{n(n+1)}{2}$

PS: Unless explicitly asked for asymptotic bound, we should always try to get the exact answer.

29 votes -- kirti singh (2.6k points)

1.15.2 Recurrence: GATE CSE 1988 | Question: 13iv top

https://gateoverflow.in/94637

✔ $T(n) = T(n/2) + 1$

$= T(n/4) + 2$

$= T(n/8) + 3$

$\vdots$

$= T(n/2^k) + k.$

Recurrence stops when $2^k >= n$.

When $2^k = n, k = \lg n$

So, $T(n) = T(1) + \lg n = 1 + \lg n$

PS: Unless explicitly asked for asymptotic bound, we should give exact answers for solutions of recurrence equations.

15 votes -- Arjun Suresh (330k points)

1.15.3 Recurrence: GATE CSE 1989 | Question: 13b top https://gateoverflow.in/93175

✔ $T(n) = T(\frac{n}{2}) + \sqrt{n}$

$= T(\frac{n}{4}) + \sqrt{n} + \sqrt{(n/2)}$

$\vdots$

$= \sqrt{n} + \sqrt{(n/2)} + \sqrt{(n/4)} + \sqrt{(n/8)} + \ldots + \sqrt{(n/2^{\lg n-1})} + T(1)$

$= \sqrt{n}\left[1 + \frac{1}{\sqrt{2}} + \frac{1}{\sqrt{2}^2} + \ldots + \frac{1}{\sqrt{2}^{\lg n}}\right]$

$= \sqrt{n}\left[\frac{1-(\frac{1}{\sqrt{2}})^{\lg n+1}}{1-\frac{1}{\sqrt{2}}}\right]$ (Sum of $\lg n + 1$ terms of GP with $a = 1$ and $r = 1\sqrt{2}$)

$= \sqrt{n}\left[\frac{1-\frac{1}{\sqrt{2}\sqrt{n}}}{1-\frac{1}{\sqrt{2}}}\right]$

$= \frac{\sqrt{2n}-1}{\sqrt{2}-1}$

$= \left(\sqrt{2n} - 1\right)\left(\sqrt{2} + 1\right)$

$= \sqrt{n}\left(2 + \sqrt{2}\right) - \sqrt{2} - 1$

16 votes -- Arjun Suresh (330k points)

1.15.4 Recurrence: GATE CSE 1990 | Question: 17a top https://gateoverflow.in/86878

✔ $T(n) = \frac{n+1}{n}T(n-1) + 1 \qquad \rightarrow (1)$

$T(n-1) = \frac{n}{n-1}T(n-2) + 1 \qquad \rightarrow (2)$

$T(n-2) = \frac{n-1}{n-2}T(n-3) + 1 \qquad \rightarrow (3)$

Substituting value of $T(n-1)$ from (2) in (1)

$\implies T(n) = \frac{n+1}{n} * \frac{n}{n-1}T(n-2) + \frac{n+1}{n} + 1$

$\implies T(n) = \frac{n+1}{n-1}T(n-2) + \frac{n+1}{n} + 1$

Now substituting value of $T(n-2)$ in above equation

$\implies T(n) = \frac{n+1}{n-1} * \frac{n-1}{n-2}T(n-3) + \frac{n+1}{n-1} + \frac{n+1}{n} + 1$

$\implies T(n) = \frac{n+1}{n-2}T(n-3) + \frac{n+1}{n-1} + \frac{n+1}{n} + 1$

$\vdots$

so on

$T(n) = \frac{n+1}{n-k+1}T(n-k) + \frac{n+1}{n} + \frac{n+1}{n-1} + \ldots + \frac{n+1}{n-k+2} + 1$

$T(n) = \frac{n+1}{n-k+1}T(n-k) + (n+1) * (\frac{1}{n} + \frac{1}{n-1} + \ldots + \frac{1}{n-k+2}) + 1$

Now let $n - k = 1$ so $k = n - 1$, substitute value of k in above equation

$$\implies T(n) = \frac{n+1}{n-(n-1)+1}T(1) + (n+1) * (\frac{1}{n} + \frac{1}{n-1} + \ldots + \frac{1}{n-(n-1)+2}) + 1$$

$$\implies T(n) = \frac{n+1}{2} + (n+1) \times (\frac{1}{n} + \frac{1}{n-1} + \ldots + \frac{1}{3}) + 1$$

$$\implies T(n) = \frac{n+1}{2} + (n+1) \times (H_n - \frac{1}{2} - 1) + 1$$

$$\implies T(n) = \frac{n+1}{2} + (n+1) \times H_n - \frac{n+1}{2} - (n+1) + 1$$

$$\implies \mathbf{T(n) = (n+1) \times H_n - n}$$

Now, $H_n \approx \log n + \gamma$

where γ is the Euler-Mascheroni constant.

$$\mathbf{T(n) = O(n \log n)}$$

33 votes -- Digvijaysingh Gautam (6.3k points)

1.15.5 Recurrence: GATE CSE 1992 | Question: 07a top

https://gateoverflow.in/586

1. The function $F(n)$ is NOT a recursive function. You can't have a recurrence relation for it in the first place!
2. $F(n)$ calculates $(n-1)^n$.

The equivalent C++ code is as follows: (You can try it out here: http://ideone.com/w0u4lk)

```
long F(long n) {
 long F1 = 1;

 if(n==1) { return 3; }
 else {
  for(long i = 1; i <= n; i++) {
   long C = 0;
   // Note: the belore For loop only has one line
   for(long j = 1; j <= n-1; j++) { C = C+1; }
   // At the end of this for loop, C will be = (n-1)
   F1 = F1 * C;
  }
 }
 return F1;
}
```

It is clear that the inner for loop can be replaced by a single statement as follows:

```
long F(long n) {
 long F1 = 1;

 if(n==1) { return 3; }
 else {
  for(long i = 1; i <= n; i++)
   F1 = F1 * (n-1);
 }
 return F1;
}
```

And this calculates $(n-1)^n$

References

33 votes -- Pragy Agarwal (18.3k points)

1.15.6 Recurrence: GATE CSE 1992 | Question: 07b top

https://gateoverflow.in/43600

```
   Function F(n)
begin
F1 ← 1
if(n=1) then F ← 3  //if (n==1) then return 3
```

```
else
  For i = 1 to n do
       begin
           C ← 0
          For j = 1 to n - 1 do //inner loop runs n-1 times outer loop runs for n times
          begin C ← C + 1 end   //means C=n-1
          F1 = F1 * C  //means n-1 is getting multiplied n times so ans is (n-1)^n for n>=2
      end
   F = F1
end
```

14 votes -- Rajesh Pradhan (18.9k points)

1.15.7 Recurrence: GATE CSE 1993 | Question: 15 top https://gateoverflow.in/2312

✔ $a_n = a_{n-1} + n - 1$ ($n-1$ comparisons for n numbers)

$a_n = a_{n-2} + (n-2) + (n-1)$

$a_n = a_{n-3} + (n-3) + (n-2) + (n-1)$
.
.
.

$a_n = a_{n-n} + (n-n) + (n-(n-1)) + \ldots + (n-3) + (n-2) + (n-1)$

$a_n = 0 + 1 + 2 + \ldots + (n-3) + (n-2) + (n-1)$

which given $a_n = \frac{(n-1)\times(n)}{2}$

27 votes -- Rajarshi Sarkar (27.8k points)

1.15.8 Recurrence: GATE CSE 1994 | Question: 1.7, ISRO2017-14 top https://gateoverflow.in/2444

✔ Correct Option: **B**

Searching for only one half of the list. leading to $T(n/2)+$ constant time in comparing and finding mid element.

40 votes -- Gate Keeda (15.9k points)

1.15.9 Recurrence: GATE CSE 1996 | Question: 2.12 top https://gateoverflow.in/2741

✔ Answer: **A**

According to Master theorem,
$T(n) = aT(\frac{n}{b}) + f(n)$ can be expressed as:
$T(n) = [n^{\log_b a}][T(1) + u(n)]$
where $u(n) = \Theta(h(n))$ where $h(n) = \frac{f(n)}{n^{\log_b a}} = \frac{n}{n^{\log_4 3}} = n^{1-\log_4 3}$ as $h(n) = n^r$ where $r > 0$.
So, $T(n) = [n^{\log_b a}][T(1) + u(n)] = T(n) = [n^{\log_4 3}][T(1) + \Theta(n^{1-\log_4 3})] = \Theta(n^1)$.

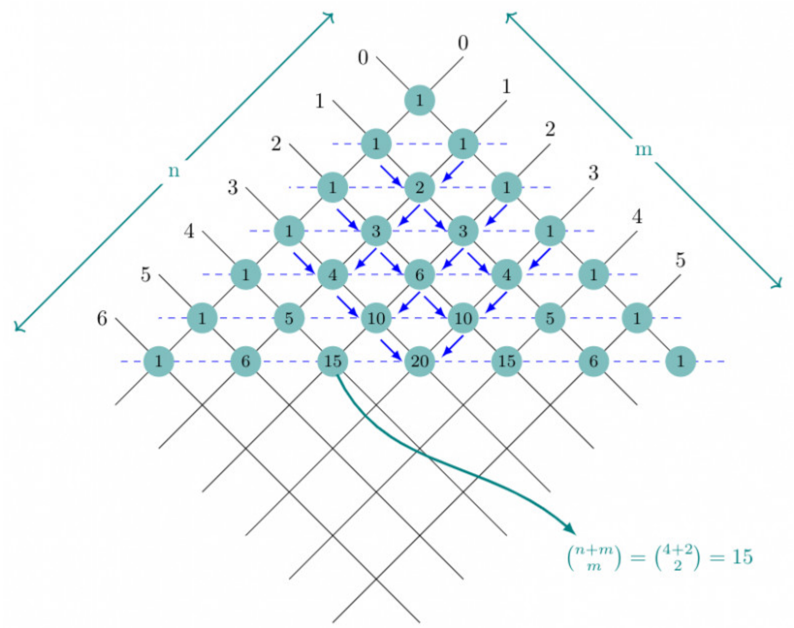
21 votes -- Rajarshi Sarkar (27.8k points)

1.15.10 Recurrence: GATE CSE 1997 | Question: 15 top https://gateoverflow.in/2275

✔ $F(n, m) = F(n-1, m) + F(n, m-1)$

Let vertices represent $F(n, m)$ value.

According to rule, we get value of $F(n, m)$ by adding value of vertex left above of it and vertex right above of it.



When counting these, we observe a pattern. Hence to find say $F(4,2)$ we do $F(4,1) + F(3,2) = 5 + 10 = 15$.

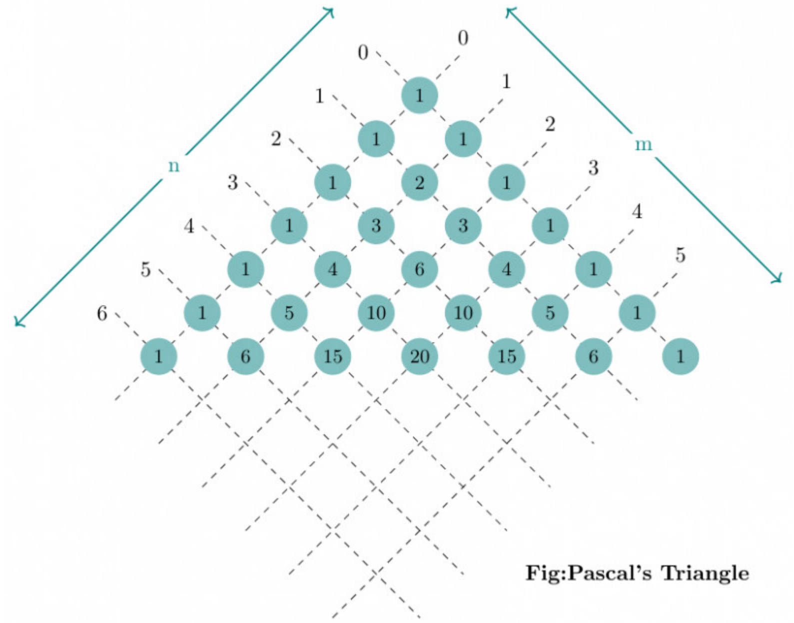


Fig:Pascal's Triangle

If $C(n, m)$ denotes number of recursive calls for $F(n, m)$ we have,
$C(n, m) = C(n-1, m) + C(n-1, m-1) + 2$

5 votes -- ARKA HALDI (353 points)

1.15.11 Recurrence: GATE CSE 1997 | Question: 4.6 top

☛ https://gateoverflow.in/2247

✔ Answer is B.

using master method (case 1)

where a = 2, b = 2

$O(n^{1/2}) < O(n^{\log_b a})$

$O(n^{1/2}) < O(n^{\log_2 2})$

31 votes -- Ankit Rokde (6.9k points)

1.15.12 Recurrence: GATE CSE 1998 | Question: 6a top

☛ https://gateoverflow.in/44584

✔ $T(n) = 2T(n-1) - 1$

$$= 2(2T(n-2)-1)-1$$

$$= 2^2T(n-2)-2-1$$

$$= 2^2(2T(n-3)-1)-2-1$$

$$= 2^3T(n-3)-2^2-2-1$$

$$\dots$$

$$= 2^{n-1}T(n-(n-1)) - \left(2^{n-2}+2^{n-3}+\dots+2^2+2+1\right)$$
$$= 2^{n-1}\times 2 - \frac{2^{n-1}-1}{2-1} \because T(1)=2, S_n = \frac{a.(r^n-1)}{r-1}$$

$$= 2^n - \left(2^{n-1}-1\right)$$

$$= 2^{n-1}+1$$

34 votes -- srestha (85.2k points)

1.15.13 Recurrence: GATE CSE 2002 | Question: 1.3 top

https://gateoverflow.in/807

✔ Let $x = 2^k$
$T(x) = 3T\left(\frac{x}{2}\right)+1$
We can apply Master's theorem case 1 with $a=3$ and $b=2$ as $f(x)=1=O\left(x^{\log_2 3-\epsilon}\right), \epsilon > 0$

So, $T(x) = \Theta\left(x^{\log_2 3}\right) = \Theta\left(2^{k\log_2 3}\right) = \Theta\left(2^{\log_2 3^k}\right) = \Theta\left(3^k\right)$

So, only option possible is **B**.

We can also directly solve as follows:

$$\begin{aligned}
T(x) &= 3T\left(\tfrac{x}{2}\right)+1\\
&= 9T\left(\tfrac{x}{4}\right)+1+3\\
&\ \vdots\\
&= 3^{\log_2 2^k} + \left(1+3+9+\dots+3^{\log_2 2^k-1}\right)\\
&\left(\text{recursion depth is } \log_2 x \text{ and } x=2^k\right)\\
&= 3^k + \tfrac{3^{\log_2 2^k}-1}{3-1}\\
&(\text{Sum to n terms of GP with } a=1 \text{ and } r=3)\\
&= 3^k + \tfrac{3^k-1}{2}\\
&= \tfrac{3.3^k-1}{2}\\
&= \tfrac{3^{k+1}-1}{2}
\end{aligned}$$

OR

$$\begin{aligned}
T\left(2^k\right) &= 3T\left(2^{k-1}\right)+1\\
&= 3^2T\left(2^{k-2}\right)+1+3\\
&\ \vdots\\
&= 3^kT\left(2^{k-k}\right)+\left(1+3+9+\dots+3^{k-1}\right)\\
&(\text{recursion depth is } k)\\
&= 3^k + \tfrac{3^{k-1}}{3-1}\\
&(\text{Sum to n terms of GP with } a=1 \text{ and } r=3)\\
&= 3^k + \tfrac{3^k-1}{2}\\
&= \tfrac{3.3^k-1}{2}\\
&= \tfrac{3^{k+1}-1}{2}
\end{aligned}$$

64 votes -- Arjun Suresh (330k points)

1.15.14 Recurrence: GATE CSE 2002 | Question: 2.11 top

☛ https://gateoverflow.in/841

✔ The complexity will be the number of times the recursion happens which is equal to the number of times we can take square root of n recursively, till n becomes 2.

$T(n) = T\left(\lceil \sqrt{n} \rceil\right) + 1$

$T(2) = 1$
$T\left(2^2\right) = T(2) + 1 = 2$
$T\left(2^{2^2}\right) = T(4) + 1 = 3$
$T\left(2^{2^3}\right) = T(16) + 1 = 4$

So, $T(n) = \lg \lg n + 1 = O\left(\log \log n\right)$

Answer : **Option C**

56 votes -- Arjun Suresh (330k points)

1.15.15 Recurrence: GATE CSE 2003 | Question: 35 top

☛ https://gateoverflow.in/925

✔ $T(m^2) = T(m^2-1) + \left\lfloor \sqrt{(m^2)} \right\rfloor$

$$= T(m^2-2) + \left\lfloor \sqrt{(m^2-1)} \right\rfloor + \left\lfloor \sqrt{(m^2)} \right\rfloor$$

$$= T(m^2-3) + \left\lfloor \sqrt{(m^2-2)} \right\rfloor + \left\lfloor \sqrt{(m^2-1)} \right\rfloor + \left\lfloor \sqrt{(m^2)} \right\rfloor$$

$$\vdots$$

$$= T(1) + \left\lfloor \sqrt{(2)} \right\rfloor + \left\lfloor \sqrt{(3)} \right\rfloor + \ldots + \left\lfloor \sqrt{(m^2)} \right\rfloor$$

$$= 3 \times 1 + 5 \times 2 + \ldots + (2m-1) \times (m-1) + m$$

(We are taking floor of square root of numbers, and between successive square roots number of numbers are in the series $3, 5, 7 \ldots$ like 3 numbers from $1..4$, 5 numbers from $5-9$ and so on).

We can try out options here or solve as shown at end:

Put $m = 5$, $T(25) = 3 \times 1 + 5 \times 2 + 7 \times 3 + 9 \times 4 + 5 = 75$

A. 59
B. 75
C. non-integer
D. 297.5

So, answer must be **B**.

$$\begin{aligned} T(m^2) &= 3 \times 1 + 5 \times 2 + \cdots + (2m-1) \times (m-1) + m \\ &= m + \sum_{i=1}^{m-1} \left[(2i+1).(i)\right] \\ &= m + \sum_{i=1}^{m-1} \left[2i^2 + i\right] \\ &= m + \tfrac{(m-1).m.(2m-1)}{3} + \tfrac{(m-1)m}{2} \\ &= \tfrac{m}{6}\left(6 + 4m^2 - 2m - 4m + 2 + 3m - 3\right) \\ &= \tfrac{m}{6}\left(4m^2 - 3m + 5\right) \end{aligned}$$

- Sum of the first n natural numbers $= \frac{n.(n+1)}{2}$.
- Sum of the squares of first n natural numbers $= \frac{n.(n+1).(2n+1)}{6}$.

51 votes -- Arjun Suresh (330k points)

1.15.16 Recurrence: GATE CSE 2004 | Question: 83, ISRO2015-40 top

https://gateoverflow.in/1077

✔ **Answer is (D)** $O(2^n)$

```
int recursive (int n) {
    if(n == 1)      // takes constant time say 'A' time
        return (1); // takes constant time say 'A' time
    else
        // takes T(n-1) + T(n-1) time
        return (recursive (n-1) + recursive (n-1));
}
```

$T(n) = 2T(n-1) + a$ is the recurrence equation found from the pseudo code. Note: a is a constant $O(1)$ cost that the non-recursive part of the function takes.

Solving the recurrence by Back Substitution:

$$\begin{aligned} T(n) &= 2T(n-1) + a \\ T(n-1) &= 2T(n-2) + a \\ T(n-2) &= 2T(n-3) + a \\ &\vdots \end{aligned}$$

Thus, we can re-write the equation for $T(n)$ as follows

$$\begin{aligned} T(n) &= 2\left[2T(n-2) + a\right] + a & &= 4T(n-2) + 2a + a \\ &= 4\left[2T(n-3) + a\right] + 3a & &= 8T(n-3) + 4a + 2a + a \\ &\vdots \\ &= 2^k T(n-k) + (2^k - 1)a \end{aligned}$$

On Substituting Limiting Condition

$T(1) = 1 \implies n - k = 1 \implies k = n - 1$

Therefore, our solution becomes

$$2^{n-1} + \left(2^{n-1} - 1\right) a = O(2^n)$$

36 votes -- Akhil Nadh PC (16.5k points)

1.15.17 Recurrence: GATE CSE 2004 | Question: 84 top

https://gateoverflow.in/1078

✔ $T(n) = 2T(n-1) + n, n \geqslant 2, T(1) = 1$

$$\begin{aligned} T(n) &= n + 2(n-1) + 2^2(n-2) + \cdots + 2^{(n-1)}(n-(n-1)) \\ &= n(1 + 2 + \cdots + 2^{n-1}) - (1.2 + 2.2^2 + 3.2^3 + \cdots + (n-1).2^{n-1}) \\ &= n(2^n - 1) - (n.2^n - 2^{n+1} + 2) \\ &= 2^{n+1} - n - 2 \end{aligned}$$

Correct Answer: A

50 votes -- suraj (4.8k points)

T(1) = 1

T(2) = 4

T(3) = 11

T(4) = 26

T(5) = 57

T(6) = 120

T(7) = 247

So,

$$T(n) = 2^{n+1} - n - 2$$

80 votes -- Arjun Suresh (330k points)

1.15.18 Recurrence: GATE CSE 2006 | Question: 51, ISRO2016-34 top https://gateoverflow.in/1829

✔ $$\begin{aligned} T(n) &= 2T\left(n^{\frac{1}{2}}\right) + 1 \\ &= 2\left(2T\left(n^{\frac{1}{2^2}}\right) + 1\right) + 1 \\ &= 4 \times T\left(n^{\frac{1}{2^2}}\right) + 5 \\ &= 8 \times T\left(n^{\frac{1}{2^3}}\right) + 13 \cdots \\ &= 2^{(\lg \lg n)} + 2 \times \lg \lg n + 1 \text{ (Proved below)} \\ &= \Theta(\lg n) \end{aligned}$$

$n^{\frac{1}{2^k}} = 2(\text{Putting 2 so that we can take log.One more step of recurrence can't change the complexity.}) \implies \frac{1}{2^k} \lg n$

So, answer is **B**, $T(n) = \Theta(\log n)$

49 votes -- Arjun Suresh (330k points)

1.15.19 Recurrence: GATE CSE 2008 | Question: 78 top https://gateoverflow.in/497

✔ $0\ 1 - 2$
$01\ 10\ 11 - 3$
$010\ 011\ 101\ 110\ 111 - 5$
$0101\ 0110\ 0111\ 1010\ 1011\ 1101\ 1110\ 1111 - 8$

So, $x_n = x_{n-1} + x_{n-2}$ (For all the strings ending in 1, we get two new strings ending in either 0 or 1 and for all strings ending in 0, we get a new string ending in 1. So, the new set of strings for x_n, will have exactly x_{n-1} strings ending in 1 and x_{n-2} strings ending in 0)

$x_5 = 8 + 5 = 13$

Correct Answer: D

31 votes -- Arjun Suresh (330k points)

1.15.20 Recurrence: GATE CSE 2008 | Question: 79 top https://gateoverflow.in/43485

✔ We can write the recurrence for the number of binary strings of length n without consecutive 0s as

- $T(n) = T(n-1) + T(n-2); \qquad n > 2$
- $T(1) = 2; T(2) = 3$

The reason for this recurrence can be seen as follows:

- A string of length n without "00" can be formed by adding
  - a "1" to any string without "00" of length $n-1 \quad \rightarrow T(n-1)$
  - a "0" to any string ending in "1" and without "00" of length $n-1 \quad \rightarrow T(n-2)$
  - These two cases are mutually exclusive (no common strings) and exhaustive (no strings outside these two cases). So, we can just add the two cases to get $T(n)$

$$\begin{aligned}
&T(1) = 2 \quad - \quad \{0,1\}\\
&T(2) = 3 \quad - \quad \{01,10,11\}\\
&T(3) = T(1) + T(2) = 2 + 3 = 5\\
&T(4) = T(3) + T(2) = 5 + 3 = 8\\
&T(5) = T(4) + T(3) = 8 + 5 = 13
\end{aligned}$$

Hence, answer is **13.**

No option matches.

3 votes -- gatecse (62.6k points)

1.15.21 Recurrence: GATE CSE 2009 | Question: 35 top https://gateoverflow.in/1321

✔ $a = 1, b = 3, \log_b a = 0$

So $n^{\log_b a} = n^0 = 1$

$f(n) = n$

So, $f(n) = \Omega(1)$

To, check Master theorem case 3, we need $c > 0$,

$f(n/3) \leq cf(n)$

$c = 1$

So using case three of master theorem

$T(n) = \Theta(f(n)) = \Theta(n)$

answer is **A.**

23 votes -- Pooja Palod (24.1k points)

1.15.22 Recurrence: GATE CSE 2012 | Question: 16 top https://gateoverflow.in/48

✔ Recurrence relation for ***Towers of Hanoi*** is

$T(1) = 1$

$T(n) = 2T(n-1) + 1$

So Answer should be **(D)**

35 votes -- Narayan Kunal (307 points)

1.15.23 Recurrence: GATE CSE 2014 Set 2 | Question: 13 top https://gateoverflow.in/1968

✔ $f(n) = \log n$

$a = 2, b = 2 \implies n^{\log_b a} = n$

So, $f(n) = \log n = O\left(n^{1-\epsilon}\right)$, we can take any ϵ from 0-1 for example 0.5 which gives $\log n = O(\sqrt(n))$, whose proof is given here: http://math.stackexchange.com/questions/145739/prove-that-logn-o-sqrtn

So, Master theorem Case 1, and answer will be $O\left(n^{\log_2 2}\right) = O(n)$

Alternate way:

$T(1) = 1T(2) = 2T(1) + \log 2 = 3 = 3n - 2T(4) = 2T(2) + \log 4 = 8 = 3n - 4T(8) = 2T(4) + \log 8 = 19 = 3n - 5$

The second term being subtracted is growing at a lower rate than the first term. So, we can say $T(n) = O(n)$.

Correct Answer: A

References

36 votes -- Arjun Suresh (330k points)

1.15.24 Recurrence: GATE CSE 2015 Set 1 | Question: 49 top https://gateoverflow.in/8355

✔ Counting the number of bit strings NOT containing two consecutive 1's. (It is easy to derive a recurrence relation for the NOT case as shown below)

$0 \quad 1$
$00 \quad 01 \quad 10 - 3$ (append both 0 and 1 to any string ending in 0, and append 0 to any string ending in 1)
$000 \quad 001 \quad 010 \quad 100 \quad 101 - 5$ (all strings ending in 0 give two strings and those ending in 1 give 1 string)
$0000 \quad 0001 \quad 0010 \quad 0100 \quad 0101 \quad 1000 \quad 1001 \quad 1010 - 8$
$\vdots$

$a'_n = a'_{n-1} + a'_{n-2}$ (where a_n denote the number of bit strings of length n containing two consecutive 1s)

$2^n - a_n = (2^{n-1} - a_{n-1}) + (2^{n-2} - a_{n-2})$

$a_n = 2^{n-2}(4 - 2 - 1) + a_{n-1} + a_{n-2}$

$a_n = a_{n-1} + a_{n-2} + 2^{n-2}$

Correct Option: **A**

78 votes -- Arjun Suresh (330k points)

1.15.25 Recurrence: GATE CSE 2015 Set 3 | Question: 39 top https://gateoverflow.in/8498

✔ **Answer : Option B**

$\text{T(n)} = \text{T(n-1)} + \text{T(n-3)} + 2$, here T(n) denotes the number of times a recursive call is made for input n. 2 denotes the two direct recursive calls.

$\text{T}(\text{n} \leq 0) = 0$
$\text{T}(1) = 2$
$\text{T}(2) = 4$
$\text{T}(3) = 6$
$\text{T}(4) = 10$
$\text{T}(5) = 16$
$\text{T}(6) = 24$

So, answer is $24 + 1$ call from main $= 25$.

67 votes -- Arjun Suresh (330k points)

1.15.26 Recurrence: GATE CSE 2016 Set 2 | Question: 39 top https://gateoverflow.in/39581

✔ If they are asking for worst case complexity hence,
By calling $A(n)$ we get $A(n/2)$ 5 times,

$$A(n) = 5A(n/2) + O(1)$$

Hence, by applying masters theorem,
Case 1 : $a > b^k$

$n^{\log_2 5}$

Thus value of alpha will be 2.32

83 votes -- Shashank Chavan (2.4k points)

1.15.27 Recurrence: GATE CSE 2017 Set 2 | Question: 30 top https://gateoverflow.in/118623

✔ $T(n) = 2T(\sqrt{n}) + 1$

Put, $n = 2^m$

$T(2^m) = 2T(2^{m/2}) + 1$

put, $T(2^m) = s(m)$

$s(m) = 2s(m/2) + 1$

Using case 1 of master method ,

$= \Theta(m) = \Theta(\log n)$

https://gateoverflow.in/1829/gate2006-51-isro2016-34?show=37791#c37791

Correct Answer: B

References

52 votes -- jatin saini (4.2k points)

1.15.28 Recurrence: GATE CSE 2020 | Question: 2 top

https://gateoverflow.in/333229

✔ $$T(n) = \begin{cases} T\left(n^{1/a}\right) + 1 & ;\text{when } n \neq b \\ 1 & ;\text{when } n = b \end{cases}$$

Now, $T(n) = T\left(n^{1/a}\right) + 1$

$$= T\left(n^{1/a^2}\right) + 1 + 1 \quad \left[\because T\left(n^{1/a}\right) = T\left(n^{1/a^2}\right) + 1\right]$$

$$= T\left(n^{1/a^2}\right) + 1 + 1 + 1 \quad \left[\because T\left(n^{1/a^2}\right) = T\left(n^{1/a^3}\right) + 1\right]$$

After k iterations, $T(n) = T\left(n^{1/a^k}\right) + k$

When $n^{1/a^k} = b, i.e.,\ \frac{1}{a^k} \log n = \log b$

$$\implies a^k = \frac{\log n}{\log b}$$

$$\implies k = \log_a \log_b n$$

[$\because$ $a\&b$ are $\omega(1)$, so $a\&b$ are some function of n and not constant. So $a\&b$ can't be replaced with 2]

So, option D is rejected.

Now, $T(n) = T\left(n^{1/a^k}\right) + k$

$$= T(b) + \log_a \log_b n$$

$$= 1 + \log_a \log_b n$$

$$= \Theta\left(\log_a \log_b n\right)$$

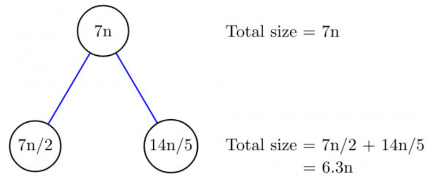
So, the correct answer is A.

41 votes -- Aditya Jain (303 points)

1.15.29 Recurrence: GATE CSE 2021 Set 1 | Question: 30 top ☛ https://gateoverflow.in/357421

✔ **Correct Option: C**

At first, draw two levels of the recursion tree:



Size of the problem will reduce as we go down a level each time in the tree.

Hence, in this case root will be the dominant part.

Final ans is $T(n) = \Theta(n)$ (order of the root) (you have to ans in less than 30 sec!)

Let $f(n) = 1 + a + a^2 + \ldots + a^n, a < 1 \implies f(n) = \frac{1-a^{(n+1)}}{1-a}$

if $a < 1$, then $a^{(n+1)} < a \implies f(n) < 1 \implies f(n) = O(1)$

We can see that the size of problem reduces geometrically, and root is the dominating part.

Note: Apply the following method for **divide and conquer recursions** because the size of the problem reduces **geometrically**. Not for **subtract and conquer** where it might reduce **arithmetically!**

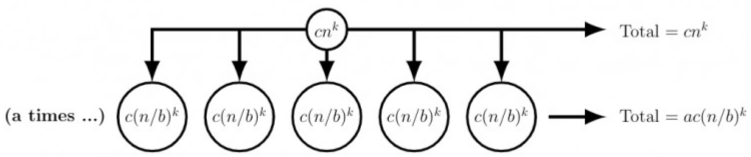
And to apply this idea, all the conditions of Master theorem must be satisfied.

Everything given below is not required for GATE, it's just to explain the idea.

$T(n) = aT\left(\frac{n}{b}\right) + \Theta(n^k)$, $n > 1$

$T(1) = c$

(Assume all conditions of the Master theorem to be true here, $b > 1$).



Let height of this tree be h,

$\frac{n}{b^h} = 1 \implies b^h = n \implies h = \log_b(n)$

No. of leaves $= \underbrace{a * a * a * \ldots * a}_{\text{times the height of the tree}} = a^{\log_b(n)} = n^{\log_b(a)}$

Here's the relation with master theorem:

Case 1 (root is dominant): $cn^k > ac\left(\frac{n}{b}\right)^k$

$\implies 1 > \frac{a}{b^k} \implies b^k > a \implies \boldsymbol{k > \log_b(a)}$

$\implies n^k > n^{\log_b(a)}$

This case of Master theorem says $T(n) = \Theta(n^k)$, which is nothing but the order of root.

(We are kinda deriving cases of the master theorem using intuition)

Case 2 (root is equal to all levels): $cn^k = ac\left(\frac{n}{b}\right)^k$

$\implies 1 = \frac{a}{b^k} \implies b^k = a \implies \boldsymbol{k = \log_b(a)}$

$\implies n^k = n^{\log_b(a)}$

According to the Master theorem, $T(n) = \Theta(n^k \log(n))$.

All the levels are equally dominant, hence final result will be the sum of all levels. That is,

size of a particular level $\times$ the height $\implies n^k \times \log_b(n)$, Hence $T(n) = \Theta(n^k \log_b(n))$.

Case 3 (leaf level is dominant):

$\implies 1 < \frac{a}{b^k} \implies b^k < a \implies \boldsymbol{k < \log_b(a)}$

$\implies n^k < n^{\log_b(a)}$

According to Master theorem, $T(n) = \Theta(n^{\log_b(a)})$

As the leaves' levels are dominant, Ans will be in the order of the leaf's level = Order of no of leaves in the tree (because base condition will be a constant)

Hence $T(n) = \Theta(n^{\log_b(a)})$

After reading this logic you'll never have to remember the Master theorem!

Reference

You'll find different kind of recurrence relations and how to solve them here:

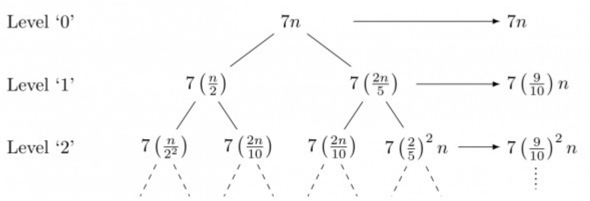
- https://jeffe.cs.illinois.edu/teaching/algorithms/notes/99-recurrences.pdf

References

2 votes -- Nikhil Dhama (2.4k points)

$$T(n) = T\left(\frac{n}{2}\right) + T\left(\frac{2n}{5}\right) + 7n$$

Recursion Tree would look like :



Since, $\frac{n}{2} > \frac{2n}{5}$, So, the height of the rightmost subtree will determine the lower bound of the given recurrence, and the height of the leftmost subtree will determine the upper bound of the given recurrence.

Height of the leftmost subtree is : $\log_2 n$ and so,

$T(n) \leq 7n + 7\left(\frac{9}{10}\right)n + 7\left(\frac{9}{10}\right)^2 n + \ldots + 7\left(\frac{9}{10}\right)^{\log_2 n} n$

$\implies T(n) \leq 7n\left(1 + \frac{9}{10} + \left(\frac{9}{10}\right)^2 + \ldots + \left(\frac{9}{10}\right)^{\log_2 n}\right)$

$\implies T(n) \leq 7n\left(\frac{1-\left(\frac{9}{10}\right)^{\log_2 n+1}}{1-\frac{9}{10}}\right) = 70n\left(1-\left(\frac{9}{10}\right)^{\log_2 n+1}\right)$

$= 70n - 70n\left(n^{\log_2 \frac{9}{10}} \frac{9}{10}\right) = 70n - 63n^{0.85}$

$\implies T(n) \leq 70n$

$\implies T(n) \in O(n)$

Height of the rightmost subtree is : $\log_{5/2} n$ and so,

$T(n) \geq 7n + 7\left(\frac{9}{10}\right)n + 7\left(\frac{9}{10}\right)^2 n + \ldots + 7\left(\frac{9}{10}\right)^{\log_{5/2} n} n$

$\implies T(n) \geq 7n\left(1 + \frac{9}{10} + \left(\frac{9}{10}\right)^2 + \cdots + \left(\frac{9}{10}\right)^{\log_{5/2}} n\right)$

$\implies T(n) \geq 7n\left(\frac{1-\left(\frac{9}{10}\right)^{\log_{5/2}(n+1)}}{1-\frac{9}{10}}\right) = 70n\left(1-\left(\frac{9}{10}\right)^{\log_{5/2} n+1}\right)$

$= 70n - 70n\left(n^{\log_{5/2} \frac{9}{10}} \frac{9}{10}\right) = 70n - 63n^{0.89}$

$\implies T(n) \geq n\left(70 - \frac{63}{n^{0.11}}\right)$

$\because \lim_{n\to\infty} \frac{63}{n^{0.11}} = 0$ for n ≥ 1 So, $\left(70 - \frac{63}{n^{0.11}}\right)$ is a positive constant $c > 0$ for large n and so, $T(n) \geq cn$.

- $T(n) \in \Omega(n)$
- $T(n) \sim 70n$
- $T(n) \in \Theta(n)$

2 votes -- ankitgupta.1729 (15k points)

1.15.30 Recurrence: GATE CSE 2021 Set 2 | Question: 39 top https://gateoverflow.in/357501

✔ Options A and B are definitely wrong, condition on $f(n)$ can't be independent of a and b in any case, it should take both a and b into account.

Option C is correct. standard **case of master theorem**, if $f(n)$ is polynomial time smaller than $O(n^{\log_b a})$, then $T(n) = \Theta(n^{\log_b a})$. (see case 1 below).

Theorem 4.1 (Master theorem)

Let $a \geq 1$ and $b > 1$ be constants, let $f(n)$ be a function, and let $T(n)$ be defined on the nonnegative integers by the recurrence

$$T(n) = aT(n \qquad /$$

where we interpret n/b to mean either $\lfloor n/b \rfloor$ or $\lceil n/b \rceil$. Then $T(n)$ has the following asymptotic bounds:

1. If $f(n) = O(n^{\log_b a-\varepsilon})$ for some constant $\varepsilon > 0$, then $T(n) = \Theta(n^{\log_b a})$.
2. If $f(n) = \Theta(n^{\log_b a})$, then $T(n) = \Theta(n^{\log_b a} \lg n)$.
3. If $f(n) = \Omega(n^{\log_b a+\varepsilon})$ for some constant $\varepsilon > 0$, and if $af(n/b) \leq cf(n)$ for some constant $c < 1$ and all sufficiently large n, then $T(n) = \Theta(f(n))$.

Reference

Option D is wrong, (see case 2 above).

A good slide to understand master theorem and the idea behind it.

References

5 votes -- Nikhil Dhama (2.4k points)

1.15.31 Recurrence: GATE IT 2004 | Question: 57 top https://gateoverflow.in/3700

✔ Answer is **B.**

For binary search on n elements we pick a path of $n/2$ elements after doing a single comparison $(O(1))$ and discard the remaining $n/2$ elements So, $T(n) = T(n/2) + 1$.

For merge sort on n elements we divide the array into two equal parts containing $n/2$ elements, recursively merge sort them and finally merges the 2 arrays in $O(n)$ time. So, $T(n) = 2T(n/2) + cn$.

For quick sort on n elements we split the array based on the position of the pivot. If pivot happens to be the $k + 1^{th}$ element in the sorted array, we do a recursive call using k elements and $n - k - 1$ elements and we need $O(n)$ time to decide the position of the pivot. So, $T(n) = T(n - k - 1) + T(k) + cn$. (A "$-1$" is missing in the question option).

For Tower of Hanoi using n disks, to position the first disk we have to replace $n - 1$ disks two times. To place the remaining disks we have to recursively solve the problem thus giving $T(n) = 2T(n - 1) + 1$.

References

1 votes -- Arjun Suresh (330k points)

1.15.32 Recurrence: GATE IT 2005 | Question: 51 top https://gateoverflow.in/3812

✔ Option C is the answer. It can be done by Master's theorem.

$n^{\log_b a} = n^{\log_2 2} = n$.

$f(n) = \sqrt{n} = n^{\frac{1}{2}}$.

So, $f(n) = O\left(n^{\log_b a - \epsilon}\right)$ is true for any real ϵ, $0 < \epsilon < \frac{1}{2}$. Hence Master theorem Case 1 satisfied,

$$T(n) = \Theta\left(n^{\log_b a}\right) = \Theta(n).$$

31 votes -- Bhagirathi Nayak (11.7k points)

1.15.33 Recurrence: GATE IT 2008 | Question: 44 top https://gateoverflow.in/3354

✔
$$\begin{aligned} T(n) &= \sqrt(2)T\left(\frac{n}{2}\right) + \sqrt{n} \\ &= \sqrt{2}^2 T\left(\frac{n}{2^2}\right) + \sqrt{2}\sqrt{\frac{n}{2}} + \sqrt{n} \\ &\vdots \\ &= \sqrt{2}^{\lg n} T(1) + \lg n \sqrt{n} \\ &= \sqrt{n} + \lg n \sqrt{n} \\ &= \sqrt{n}\,(\lg n + 1) \end{aligned}$$

If we use Master theorem we get option **B**. But one must know that Master theorem is used to find the asymptotic bound and not an EXACT value. And in the question here it explicitly says **"evaluates to"**.

73 votes -- Arjun Suresh (330k points)

1.16 Recursion (3) top

1.16.1 Recursion: GATE CSE 1995 | Question: 2.9 top https://gateoverflow.in/2621

A language with string manipulation facilities uses the following operations

```
head(s): first character of a string
tail(s): all but exclude the first character of a string
```

```
concat(s1, s2): s1s2
```

For the string "$acbc$" what will be the output of

```
concat(head(s), head(tail(tail(s))))
```

A. ac
B. bc
C. ab
D. cc

gate1995 algorithms normal recursion

Answer

1.16.2 Recursion: GATE CSE 2018 | Question: 45 top https://gateoverflow.in/204120

Consider the following program written in pseudo-code. Assume that x and y are integers.

```
Count (x, y) {
    if (y !=1 ) {
        if (x !=1) {
            print("*");
            Count (x/2, y);
        }
        else {
            y=y-1;
            Count (1024, y);
        }
    }
}
```

The number of times that the $print$ statement is executed by the call $Count(1024, 1024)$ is _____

gate2018-cse numerical-answers algorithms recursion

Answer

1.16.3 Recursion: GATE CSE 2021 Set 2 | Question: 49 top https://gateoverflow.in/357488

Consider the following ANSI C program

```
#include <stdio.h>
int foo(int x, int y, int q)
    {
        if ((x<=0) && (y<=0))
        return q;
        if (x<=0)
        return foo(x, y-q, q);
        if (y<=0)
        return foo(x-q, y, q);
        return foo(x, y-q, q) + foo(x-q, y, q);
    }
int main( )
{
    int r = foo(15, 15, 10);
    printf("%d", r);
    return 0;
}
```

The output of the program upon execution is _________

gate2021-cse-set2 algorithms recursion output numerical-answers

Answer

Answers: Recursion

1.16.1 Recursion: GATE CSE 1995 | Question: 2.9 top https://gateoverflow.in/2621

✔ Answer is **C.**

- $concat(a, head(tail(tail(acbc))))$
- $concat(a, head(tail(cbc)))$
- $concat(a, head(bc))$
- $concat(a, b)$
- $ab.$

22 votes -- Gate Keeda (15.9k points)

1.16.2 Recursion: GATE CSE 2018 | Question: 45 top https://gateoverflow.in/204120

✔

```
Count (x, y) {
 if (y !=1 ) {
     if (x !=1) {
         print("*");
         Count (x/2, y);
     }
     else {
         y=y-1;
         Count (1024, y);
     }
 }
}
```

Here, for each y value print("$*$") will run 10 times. Once x value reaches 1, $count(1024, y-1)$ will be called. Variable y can take values $[2 \ \ 1024]$ $i.e.$ total 1023 values. So,

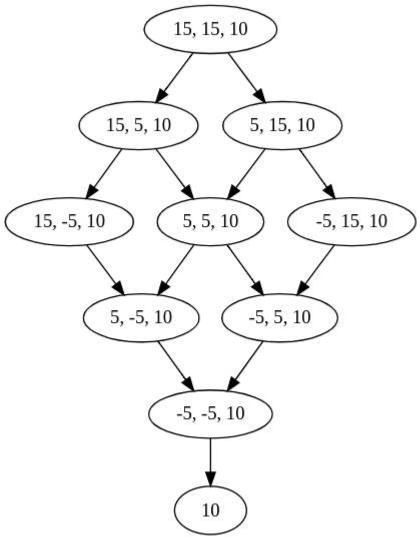
The total number of times '$*$' will be printed = $\left(\text{Number of times '}*\text{' printed per } y \text{ value}\right) * \left(\text{number of values } y \text{ takes}\right)$.

Number of times '$*$' printed $= 10 * 1023 = 10230$

25 votes -- Digvijay (44.9k points)

1.16.3 Recursion: GATE CSE 2021 Set 2 | Question: 49 top https://gateoverflow.in/357488

✔



Single child parent inherit the value of child, double child parents inherit the sum of the value of their children.

The function will return the value 60.

9 votes -- zxy123 (2.5k points)

1.17 Searching (6) top

1.17.1 Searching: GATE CSE 1996 | Question: 18 top https://gateoverflow.in/2770

Consider the following program that attempts to locate an element x in an array $a[]$ using binary search. Assume $N > 1$. The program is erroneous. Under what conditions does the program fail?

```
var i,j,k: integer; x: integer;
    a: array; [1..N] of integer;
begin i:= 1; j:= n;
repeat
    k:(i+j) div 2;
    if a[k] < x then i:= k
    else j:= k
until (a[k] = x) or (i >= j);

if (a[k] = x) then
    writeln ('x is in the array')
else
    writeln ('x is not in the array')
end;
```

gate1996 algorithms searching normal descriptive

Answer

1.17.2 Searching: GATE CSE 1996 | Question: 2.13, ISRO2016-28 top https://gateoverflow.in/2742

The average number of key comparisons required for a successful search for sequential search on n items is

A. $\frac{n}{2}$
B. $\frac{n-1}{2}$
C. $\frac{n+1}{2}$
D. None of the above

gate1996 algorithms easy isro2016 searching

Answer

1.17.3 Searching: GATE CSE 2002 | Question: 2.10 top https://gateoverflow.in/840

Consider the following algorithm for searching for a given number x in an unsorted array $A[1..n]$ having n distinct values:

1. Choose an i at random from $1..n$
2. If $A[i] = x$, then Stop else Goto 1;

Assuming that x is present in A, what is the expected number of comparisons made by the algorithm before it terminates?

A. n
B. $n - 1$
C. $2n$
D. $\frac{n}{2}$

gate2002-cse searching normal

Answer

1.17.4 Searching: GATE CSE 2008 | Question: 84 top

https://gateoverflow.in/394

Consider the following C program that attempts to locate an element x in an array $Y[\,]$ using binary search. The program is erroneous.

```
f (int Y[10] , int x) {
    int u, j, k;
    i= 0; j = 9;
    do {
        k = (i+ j) / 2;
        if( Y[k] < x) i = k;else j = k;
        } while (Y[k] != x) && (i < j)) ;
    if(Y[k] == x) printf(" x is in the array ") ;
    else printf(" x is not in the array ") ;
}
```

On which of the following contents of Y and x does the program fail?

A. Y is $[1\ 2\ 3\ 4\ 5\ 6\ 7\ 8\ 9\ 10]$ and $x < 10$
B. Y is $[1\ 3\ 5\ 7\ 9\ 11\ 13\ 15\ 17\ 19]$ and $x < 1$
C. Y is $[2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2\ 2]$ and $x > 2$
D. Y is $[2\ 4\ 6\ 8\ 10\ 12\ 14\ 16\ 18\ 20]$ and $2 < x < 20$ and x is even

gate2008-cse algorithms searching normal

Answer

1.17.5 Searching: GATE CSE 2008 | Question: 85 top

https://gateoverflow.in/43508

Consider the following C program that attempts to locate an element x in an array $Y[\,]$ using binary search. The program is erroneous.

```
f (int Y[10] , int x) {
    int u, j, k;
    i= 0; j = 9;
    do {
        k = (i+ j) / 2;
        if( Y[k] < x) i = k;else j = k;
        } while (Y[k] != x) && (i < j)) ;
    if(Y[k] == x) printf(" x is in the array ") ;
    else printf(" x is not in the array ") ;
}
```

The correction needed in the program to make it work properly is

A. Change line 6 to: if $(Y[k] < x) i = k + 1$; else $j = k - 1$;
B. Change line 6 to: if $(Y[k] < x) i = k - 1$; else $j = k + 1$;
C. Change line 6 to: if $(Y[k] < x) i = k$; else $j = k$;
D. Change line 7 to: } while $((Y[k] == x)\&\&(i < j))$;

gate2008-cse algorithms searching normal

Answer

1.17.6 Searching: GATE CSE 2017 Set 1 | Question: 48 top https://gateoverflow.in/118331

Let A be an array of 31 numbers consisting of a sequence of 0's followed by a sequence of 1's. The problem is to find the smallest index i such that $A\left[i\right]$ is 1 by probing the minimum number of locations in A. The *worst case* number of probes performed by an *optimal* algorithm is ____________.

gate2017-cse-set1 algorithms normal numerical-answers searching

Answer

Answers: Searching

1.17.1 Searching: GATE CSE 1996 | Question: 18 top https://gateoverflow.in/2770

✔ The code is wrong here

```
k=(i+j) / 2;
if (a[k] < x) then i = k;
else j = k;
```

The (correct) code should be:

```
k=(i+j) / 2;
if (a[k] < x) then i = k + 1;
else j = k - 1;
```

We can try an example with the given code in question

Let the array be $a[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]$
Index numbers $1, 2, 3, 4, 5, 6, 7, 8, 9, 10$
Let $x = 10$; now run the code;

Initially $i = 1, j = 10$;

first time $k = (i + j)/2 = 11/2 = 5.5 = 5$ (because of integer type) $= i$

second time $= k = (i + j)/2 = 15/2 = 7.5 = 7 = i$

third time $= k = (i + j)/2 = 17/2 = 8.5 = 8 = i$

fourth time $= k = (i + j)/2 = 18/2 = 9 = i$

fifth time $= k = (i + j)/2 = 19/2 = 9.5 = 9 = i$

sixth time $= k = (i + j)/2 = 19/2 = 9.5 = 9 = i$

seventh time $= k = (i + j)/2 = 19/2 = 9.5 = 9 = i$

Going to infinite loop (run time error)

For terminating the loop, it should be $i = k + 1$ instead of $i = k$ and $j = k - 1$ instead of $j = k$;

27 votes -- Mithlesh Upadhyay (4.3k points)

1.17.2 Searching: GATE CSE 1996 | Question: 2.13, ISRO2016-28 top https://gateoverflow.in/2742

✔ Expected number of comparisons
$= 1\times$ Probability of first element being $x + 2\times$ Probability of second element being $x + \ldots + n\times$ Probability of last element being x.

$= \frac{1}{n} + \frac{2}{n} + \frac{3}{n} + \ldots + \frac{n}{n}$

$= \frac{\left(\frac{n\times(n+1)}{2}\right)}{n}$

$= \frac{n+1}{2}$

Correct Answer: C

92 votes -- Arjun Suresh (330k points)

1.17.3 Searching: GATE CSE 2002 | Question: 2.10 top

https://gateoverflow.in/840

✔ Expected number of comparisons $(E) = 1\times$ Probability of find on first comparison $+2\times$Probability of find on second comparison $+\ldots+i\times$ Probability of find on ith comparison $+\ldots$

$= 1 \times \frac{1}{n} + 2 \times \frac{n-1}{n^2} + 3 \times \frac{(n-1)^2}{n^3} + \ldots$

$= \frac{1/n}{1-\frac{n-1}{n}} + \frac{(n-1)/n^2}{\left(1-\frac{n-1}{n}\right)^2}$ (Sum to infinity of aritmetico-geometric series with $a = \frac{1}{n}, r = \frac{n-1}{n}$ and $d = \frac{1}{n}) = 1 + n - 1 =$

Reference: https://en.wikipedia.org/wiki/Arithmetico-geometric_sequence

Or we can also do,

$E = 1 \times \frac{1}{n} + 2 \times \frac{n-1}{n^2} + 3 \times \frac{(n-1)^2}{n^3} + \ldots$

$E\frac{n-1}{n} = \frac{n-1}{n^2} + 2 \times \frac{(n-1)^2}{n^3} + 3 \times \frac{(n-1)^3}{n^4} + \ldots$

$E - E\frac{n-1}{n} = \frac{1}{n} + \frac{n-1}{n^2} + \frac{(n-1)^2}{n^3} + \ldots$

$E.\frac{1}{n} = \frac{(1/n)}{1-\frac{n-1}{n}} = 1$ (Sum to infinity of GP with $a = \frac{1}{n}$ and $r = \frac{n-1}{n}$) $\implies E = n$

Correct Answer: A

References

81 votes -- Arjun Suresh (330k points)

1.17.4 Searching: GATE CSE 2008 | Question: 84 top

https://gateoverflow.in/394

✔ for Q.84

when it is **option C** the control will continue to iterate as $i = 8$ and $j = 9$;
again and again i will be assigned k which itself equals 8 as $\frac{8+9}{2}$ being stored in an integer type variable, will evaluate to 8.

For option A, with $x = 9$, k will take the following values:

- 4
- 6
- 7
- $8 - y[8] = 9$, x found

For option D, with $x = 10$, k will take the following values:

- 4, $y[4] = 10$, x found

37 votes -- Amar Vashishth (25.2k points)

1.17.5 Searching: GATE CSE 2008 | Question: 85 top

https://gateoverflow.in/43508

✔ Answer should be **A.**

```
if( Y[k] < x) then i = k + 1;
```

if given element that we are searching is greater, then searching will be continued in the upper half of array

otherwise $\mathbf{j = k - 1}$;

in the lower half.

Take few case in consideration i.e.

1. All elements are same
2. Increasing order with no repetition
3. Increasing order with repetition.

26 votes -- Manoj Kumar (26.7k points)

1.17.6 Searching: GATE CSE 2017 Set 1 | Question: 48 top

https://gateoverflow.in/118331

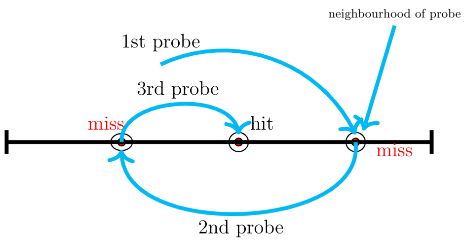
✔ Here, since 0s are followed by 1s so we have a sorted sequence and we can apply binary search.

At each stage we compare with $\frac{(low+high)}{2}^{th}$ element index and if it is 1 we check left and if it is 0 we check right.

Total worst case no. of probes is $\lceil \log_2 31 \rceil = 5$.

So, answer is 5.

👍 62 votes -- sriv_shubham (2.8k points)



It should not take more than **5 probes**

```
#include <bits/stdc++.h>
using namespace std;
#define N 100
void display(bool a[]) {
  int i=0;
  while(i<31) printf("%2d ",a[i++]);
}
void assign(bool a[],int zeros) {
  for(int i=0;i<31;i++)
    (i<zeros)?a[i] = 0:a[i] = 1;
}
int main() {
  srand(time(NULL));
  bool a[31];
  // header
  for(int i=0;i<31;i++) printf("%2d ",i);
  printf("\n\n");
  int max_probes = 0;
  for(int iteration = 1;iteration <= N;iteration++) {
    int zeros = rand()%32;
    assign(a,zeros);
    sort(a,a+31);
    int low,high,mid,ans;
    std::vector<int> seq;

    low = 0;
    high = 31;
    while(low < high) {
      mid = low + floor((high - low) / 2);
      seq.push_back(mid);
      if(a[mid] == 0) {
      low = mid + 1;
      ans = low;
        if(a[mid + 1] == 1) break;
      } else {
      high = mid;
      ans = high;
      if(mid > 0)
        if(a[mid - 1] == 0) break;

      }
    }

    display(a);
    printf(" | probes=%d  ",seq.size());
    for(auto e:seq) printf("%d ",e);
    printf(" | at = %dth\n",ans);
    //if(ans == 15) printf("\nHHH=-------------\n");
    max_probes = max(max_probes,(int)(seq.size()));
    seq.clear();
  }
  printf("%d\n",max_probes);
}
```

22 votes -- Debashish Deka (40.7k points)

1.18 Sorting (39) top

1.18.1 Sorting: GATE CSE 1988 | Question: 1iii top https://gateoverflow.in/91338

Quicksort is ________ efficient than heapsort in the worst case.

gate1988 algorithms sorting fill-in-the-blanks

Answer

1.18.2 Sorting: GATE CSE 1990 | Question: 3-v top https://gateoverflow.in/84830

The complexity of comparision based sorting algorithms is:

A. $\Theta(n \log n)$
B. $\Theta(n)$
C. $\Theta\left(n^2\right)$
D. $\Theta(n\sqrt{n})$

gate1990 normal algorithms sorting time-complexity multiple-selects

Answer

1.18.3 Sorting: GATE CSE 1991 | Question: 01,vii top https://gateoverflow.in/505

The minimum number of comparisons required to sort 5 elements is ______

gate1991 normal algorithms sorting numerical-answers

Answer

1.18.4 Sorting: GATE CSE 1991 | Question: 13 top https://gateoverflow.in/540

Give an optimal algorithm in pseudo-code for sorting a sequence of n numbers which has only k distinct numbers (k is not known a Priori). Give a brief analysis for the time-complexity of your algorithm.

gate1991 sorting time-complexity algorithms difficult descriptive

Answer

1.18.5 Sorting: GATE CSE 1992 | Question: 02,ix top https://gateoverflow.in/559

Following algorithm(s) can be used to sort n in the range $[1 \dots n^3]$ in $O(n)$ time

a. Heap sort
b. Quick sort
c. Merge sort
d. Radix sort

gate1992 easy algorithms sorting multiple-selects

Answer

1.18.6 Sorting: GATE CSE 1995 | Question: 1.16 top https://gateoverflow.in/2603

For merging two sorted lists of sizes m and n into a sorted list of size $m+n$, we require comparisons of

A. $O(m)$
B. $O(n)$
C. $O(m+n)$
D. $O(\log m + \log n)$

gate1995 algorithms sorting normal

Answer

1.18.7 Sorting: GATE CSE 1995 | Question: 12 top https://gateoverflow.in/2648

Consider the following sequence of numbers:

$$92, 37, 52, 12, 11, 25$$

Use Bubble sort to arrange the sequence in ascending order. Give the sequence at the end of each of the first five passes.

gate1995 algorithms sorting easy descriptive

Answer

1.18.8 Sorting: GATE CSE 1996 | Question: 14 top https://gateoverflow.in/2766

A two dimensional array $A[1..n][1..n]$ of integers is partially sorted if $\forall i, j \in [1..n-1], A[i][j] < A[i][j+1]$ and $A[i][j] < A[i+1][j]$

a. The smallest item in the array is at $A[i][j]$ where $i =$ __ and $j =$ __.
b. The smallest item is deleted. Complete the following $O(n)$ procedure to insert item x (which is guaranteed to be smaller than any item in the last row or column) still keeping A partially sorted.

```
procedure insert (x: integer);
var i,j: integer;
begin
    i:=1; j:=1, A[i][j]:=x;
    while (x > __ or x > __) do
        if A[i+1][j] < A[i][j] ___ then begin
            A[i][j]:=A[i+1][j]; i:=i+1;
        end
        else begin
            _____
        end
    A[i][j]:= ____
end
```

gate1996 algorithms sorting normal descriptive

Answer

1.18.9 Sorting: GATE CSE 1996 | Question: 2.15 top https://gateoverflow.in/2744

Quick-sort is run on two inputs shown below to sort in ascending order taking first element as pivot

i. $1, 2, 3, \dots n$
ii. $n, n-1, n-2, \dots, 2, 1$

Let C_1 and C_2 be the number of comparisons made for the inputs (i) and (ii) respectively. Then,

A. $C_1 < C_2$
B. $C_1 > C_2$
C. $C_1 = C_2$
D. we cannot say anything for arbitrary n

gate1996 algorithms sorting normal

Answer

1.18.10 Sorting: GATE CSE 1998 | Question: 1.22 top https://gateoverflow.in/1659

Give the correct matching for the following pairs:

(A)	$O(\log n)$	(P)	Selection sort
(B)	$O(n)$	(Q)	Insertion sort
(C)	$O(n \log n)$	(R)	Binary search
(D)	$O(n^2)$	(S)	Merge sort

A. A-R B-P C-Q D-S
B. A-R B-P C-S D-Q

C. A-P B-R C-S D-Q
D. A-P B-S C-R D-Q

gate1998 algorithms sorting easy

Answer

1.18.11 Sorting: GATE CSE 1999 | Question: 1.12 top

https://gateoverflow.in/1465

A sorting technique is called stable if

A. it takes $O(n \log n)$ time
B. it maintains the relative order of occurrence of non-distinct elements
C. it uses divide and conquer paradigm
D. it takes $O(n)$ space

gate1999 algorithms sorting easy

Answer

1.18.12 Sorting: GATE CSE 1999 | Question: 1.14, ISRO2015-42 top

https://gateoverflow.in/1467

If one uses straight two-way merge sort algorithm to sort the following elements in ascending order:

$20, 47, 15, 8, 9, 4, 40, 30, 12, 17$

then the order of these elements after second pass of the algorithm is:

A. $8, 9, 15, 20, 47, 4, 12, 17, 30, 40$
B. $8, 15, 20, 47, 4, 9, 30, 40, 12, 17$
C. $15, 20, 47, 4, 8, 9, 12, 30, 40, 17$
D. $4, 8, 9, 15, 20, 47, 12, 17, 30, 40$

gate1999 algorithms sorting normal isro2015

Answer

1.18.13 Sorting: GATE CSE 1999 | Question: 8 top

https://gateoverflow.in/1507

Let A be an $n \times n$ matrix such that the elements in each row and each column are arranged in ascending order. Draw a decision tree, which finds 1^{st}, 2^{nd} and 3^{rd} smallest elements in minimum number of comparisons.

gate1999 algorithms sorting normal descriptive

Answer

1.18.14 Sorting: GATE CSE 2000 | Question: 17 top

https://gateoverflow.in/688

An array contains four occurrences of 0, five occurrences of 1, and three occurrences of 2 in any order. The array is to be sorted using swap operations (elements that are swapped need to be adjacent).

a. What is the minimum number of swaps needed to sort such an array in the worst case?
b. Give an ordering of elements in the above array so that the minimum number of swaps needed to sort the array is maximum.

gate2000-cse algorithms sorting normal descriptive

Answer

1.18.15 Sorting: GATE CSE 2001 | Question: 1.14 top

https://gateoverflow.in/707

Randomized quicksort is an extension of quicksort where the pivot is chosen randomly. What is the worst case complexity of sorting n numbers using Randomized quicksort?

A. $O(n)$

B. $O(n \log n)$
C. $O(n^2)$
D. $O(n!)$

gate2001-cse algorithms sorting time-complexity easy

Answer

1.18.16 Sorting: GATE CSE 2003 | Question: 22 top

https://gateoverflow.in/912

The unusual $\Theta(n^2)$ implementation of Insertion Sort to sort an array uses linear search to identify the position where an element is to be inserted into the already sorted part of the array. If, instead, we use binary search to identify the position, the worst case running time will

A. remain $\Theta(n^2)$
B. become $\Theta(n(\log n)^2)$
C. become $\Theta(n \log n)$
D. become $\Theta(n)$

gate2003-cse algorithms sorting time-complexity normal

Answer

1.18.17 Sorting: GATE CSE 2003 | Question: 61 top

https://gateoverflow.in/949

In a permutation $a_1 \dots a_n$, of n distinct integers, an inversion is a pair (a_i, a_j) such that $i < j$ and $a_i > a_j$.

If all permutations are equally likely, what is the expected number of inversions in a randomly chosen permutation of $1 \dots n$?

A. $\frac{n(n-1)}{2}$
B. $\frac{n(n-1)}{4}$
C. $\frac{n(n+1)}{4}$
D. $2n[\log_2 n]$

gate2003-cse algorithms sorting normal

Answer

1.18.18 Sorting: GATE CSE 2003 | Question: 62 top

https://gateoverflow.in/43576

In a permutation $a_1 \dots a_n$, of n distinct integers, an inversion is a pair (a_i, a_j) such that $i < j$ and $a_i > a_j$.

What would be the worst case time complexity of the Insertion Sort algorithm, if the inputs are restricted to permutations of $1 \dots n$ with at most n inversions?

A. $\Theta(n^2)$
B. $\Theta(n \log n)$
C. $\Theta(n^{1.5})$
D. $\Theta(n)$

gate2003-cse algorithms sorting normal

Answer

1.18.19 Sorting: GATE CSE 2005 | Question: 39 top

https://gateoverflow.in/784

Suppose there are $\lceil \log n \rceil$ sorted lists of $\lfloor n/\log n \rfloor$ elements each. The time complexity of producing a sorted list of all these elements is: (Hint:Use a heap data structure)

A. $O(n \log \log n)$
B. $\Theta(n \log n)$
C. $\Omega(n \log n)$
D. $\Omega\left(n^{3/2}\right)$

gate2005-cse algorithms sorting normal

Answer

1.18.20 Sorting: GATE CSE 2006 | Question: 14, ISRO2011-14 top https://gateoverflow.in/975

Which one of the following in place sorting algorithms needs the minimum number of swaps?

A. Quick sort
B. Insertion sort
C. Selection sort
D. Heap sort

gate2006-cse algorithms sorting easy isro2011

Answer

1.18.21 Sorting: GATE CSE 2006 | Question: 52 top https://gateoverflow.in/1830

The median of n elements can be found in $O(n)$ time. Which one of the following is correct about the complexity of quick sort, in which median is selected as pivot?

A. $\Theta(n)$
B. $\Theta(n \log n)$
C. $\Theta(n^2)$
D. $\Theta(n^3)$

gate2006-cse algorithms sorting easy

Answer

1.18.22 Sorting: GATE CSE 2007 | Question: 14 top https://gateoverflow.in/1212

Which of the following sorting algorithms has the lowest worse-case complexity?

A. Merge sort
B. Bubble sort
C. Quick sort
D. Selection sort

gate2007-cse algorithms sorting time-complexity easy

Answer

1.18.23 Sorting: GATE CSE 2008 | Question: 43 top https://gateoverflow.in/455

Consider the Quicksort algorithm. Suppose there is a procedure for finding a pivot element which splits the list into two sub-lists each of which contains at least one-fifth of the elements. Let $T(n)$ be the number of comparisons required to sort n elements. Then

A. $T(n) \leq 2T(n/5) + n$
B. $T(n) \leq T(n/5) + T(4n/5) + n$
C. $T(n) \leq 2T(4n/5) + n$
D. $T(n) \leq 2T(n/2) + n$

gate2008-cse algorithms sorting easy

Answer

1.18.24 Sorting: GATE CSE 2009 | Question: 11 top https://gateoverflow.in/1303

What is the number of swaps required to sort n elements using selection sort, in the worst case?

A. $\Theta(n)$
B. $\Theta(n \log n)$
C. $\Theta(n^2)$

D. $\Theta(n^2 \log n)$

gate2009-cse algorithms sorting easy

Answer

1.18.25 Sorting: GATE CSE 2009 | Question: 39 top https://gateoverflow.in/1325

In quick-sort, for sorting n elements, the $(n/4)^{th}$ smallest element is selected as pivot using an $O(n)$ time algorithm. What is the worst case time complexity of the quick sort?

A. $\Theta(n)$
B. $\Theta(n \log n)$
C. $\Theta(n^2)$
D. $\Theta(n^2 \log n)$

gate2009-cse algorithms sorting normal

Answer

1.18.26 Sorting: GATE CSE 2012 | Question: 39 top https://gateoverflow.in/1762

A list of n strings, each of length n, is sorted into lexicographic order using the merge-sort algorithm. The worst case running time of this computation is

A. $O(n \log n)$
B. $O(n^2 \log n)$
C. $O(n^2 + \log n)$
D. $O(n^2)$

gate2012-cse algorithms sorting normal

Answer

1.18.27 Sorting: GATE CSE 2013 | Question: 30 top https://gateoverflow.in/1541

The number of elements that can be sorted in $\Theta(\log n)$ time using heap sort is

A. $\Theta(1)$
B. $\Theta(\sqrt{\log n})$
C. $\Theta(\frac{\log n}{\log \log n})$
D. $\Theta(\log n)$

gate2013-cse algorithms sorting normal

Answer

hot

1.18.28 Sorting: GATE CSE 2013 | Question: 6 top https://gateoverflow.in/1415

Which one of the following is the tightest upper bound that represents the number of swaps required to sort n numbers using selection sort?

A. $O(\log n)$
B. $O(n)$
C. $O(n \log n)$
D. $O(n^2)$

3 times

gate2013-cse algorithms sorting easy

Answer

1.18.29 Sorting: GATE CSE 2014 Set 1 | Question: 14 top https://gateoverflow.in/1780

Let P be quicksort program to sort numbers in ascending order using the first element as the pivot. Let t_1 and t_2 be the number of comparisons made by P for the inputs $[1\ 2\ 3\ 4\ 5]$ and $[4\ 1\ 5\ 3\ 2]$ respectively. Which one of the following holds?

A. $t_1 = 5$
B. $t_1 < t_2$
C. $t_1 > t_2$
D. $t_1 = t_2$

gate2014-cse-set1 algorithms sorting easy

Answer

1.18.30 Sorting: GATE CSE 2014 Set 2 | Question: 38 top https://gateoverflow.in/1997

Suppose P, Q, R, S, T are sorted sequences having lengths $20, 24, 30, 35, 50$ respectively. They are to be merged into a single sequence by merging together two sequences at a time. The number of comparisons that will be needed in the worst case by the optimal algorithm for doing this is ____.

use huffman, but subtract 1 every time

gate2014-cse-set2 algorithms sorting normal numerical-answers

Answer

1.18.31 Sorting: GATE CSE 2014 Set 3 | Question: 14 top https://gateoverflow.in/2048

You have an array of n elements. Suppose you implement quicksort by always choosing the central element of the array as the pivot. Then the tightest upper bound for the worst case performance is

A. $O(n^2)$
B. $O(n \log n)$
C. $\Theta(n \log n)$
D. $O(n^3)$

gate2014-cse-set3 algorithms sorting easy

Answer

1.18.32 Sorting: GATE CSE 2015 Set 1 | Question: 2 top https://gateoverflow.in/8017

Which one of the following is the recurrence equation for the worst case time complexity of the quick sort algorithm for sorting n (≥ 2) numbers? In the recurrence equations given in the options below, c is a constant.

A. $T(n) = 2T(n/2) + cn$
B. $T(n) = T(n-1) + T(1) + cn$
C. $T(n) = 2T(n-1) + cn$
D. $T(n) = T(n/2) + cn$

gate2015-cse-set1 algorithms recurrence sorting easy

Answer

1.18.33 Sorting: GATE CSE 2015 Set 2 | Question: 45 top https://gateoverflow.in/8243

Suppose you are provided with the following function declaration in the C programming language.

```
int partition(int a[], int n);
```

The function treats the first element of $a[\,]$ as a pivot and rearranges the array so that all elements less than or equal to the pivot is in the left part of the array, and all elements greater than the pivot is in the right part. In addition, it moves the pivot so that the pivot is the last element of the left part. The return value is the number of elements in the left part.

The following partially given function in the C programming language is used to find the k^{th} smallest element in an array $a[\,]$ of size n using the partition function. We assume $k \leq n$.

```
int kth_smallest (int a[], int n, int k)
{
    int left_end = partition (a, n);
    if (left_end+1==k) {
        return a[left_end];
    }
    if (left_end+1 > k) {
        return kth_smallest (___________);
    } else {
```

```
        return kth_smallest (___________);
    }
}
```

The missing arguments lists are respectively

A. $(a, \text{left\_end}, k)$ and $(a+\text{left\_end}+1, n-\text{left\_end}-1, k-\text{left\_end}-1)$
B. $(a, \text{left\_end}, k)$ and $(a, n-\text{left\_end}-1, k-\text{left\_end}-1)$
C. $(a+ \text{left\_end}+1, n-\text{left\_end}-1, k-\text{left\_end}-1)$ and $(a, \text{left\_end}, k)$
D. $(a, n-\text{left\_end}-1, k-\text{left\_end}-1)$ and $(a,\text{left\_end}, k)$

gate2015-cse-set2 algorithms normal sorting

Answer

1.18.34 Sorting: GATE CSE 2015 Set 3 | Question: 27 top

https://gateoverflow.in/8480

Assume that a mergesort algorithm in the worst case takes 30 seconds for an input of size 64. Which of the following most closely approximates the maximum input size of a problem that can be solved in 6 minutes?

A. 256
B. 512
C. 1024
D. 2018

gate2015-cse-set3 algorithms sorting

Answer

1.18.35 Sorting: GATE CSE 2016 Set 1 | Question: 13 top

https://gateoverflow.in/39660

The worst case running times of *Insertion sort* , *Merge sort* and *Quick sort*, respectively are:

A. $\Theta(n \log n)$, $\Theta(n \log n)$ and $\Theta(n^2)$
B. $\Theta(n^2)$, $\Theta(n^2)$ and $\Theta(n \log n)$
C. $\Theta(n^2)$, $\Theta(n \log n)$ and $\Theta(n \log n)$
D. $\Theta(n^2)$, $\Theta(n \log n)$ and $\Theta(n^2)$

gate2016-cse-set1 algorithms sorting easy

Answer

1.18.36 Sorting: GATE CSE 2016 Set 2 | Question: 13 top

https://gateoverflow.in/39561

Assume that the algorithms considered here sort the input sequences in ascending order. If the input is already in the ascending order, which of the following are **TRUE**?

I. Quicksort runs in $\Theta(n^2)$ time
II. Bubblesort runs in $\Theta(n^2)$ time
III. Mergesort runs in $\Theta(n)$ time
IV. Insertion sort runs in $\Theta(n)$ time

A. I and II only
B. I and III only
C. II and IV only
D. I and IV only

gate2016-cse-set2 algorithms sorting time-complexity normal ambiguous

Answer

1.18.37 Sorting: GATE CSE 2021 Set 1 | Question: 9 top

https://gateoverflow.in/357443

Consider the following array.

23	32	45	69	72	73	89	97

Which algorithm out of the following options uses the least number of comparisons (among the array elements) to sort the above array in ascending order?

A. Selection sort
B. Mergesort
C. Insertion sort
D. Quicksort using the last element as pivot

gate2021-cse-set1 algorithms sorting

Answer

1.18.38 Sorting: GATE IT 2005 | Question: 59 top

https://gateoverflow.in/3820

Let a and b be two sorted arrays containing n integers each, in non-decreasing order. Let c be a sorted array containing $2n$ integers obtained by merging the two arrays a and b. Assuming the arrays are indexed starting from 0, consider the following four statements

I. $a[i] \geq b[i] \Rightarrow c[2i] \geq a[i]$
II. $a[i] \geq b[i] \Rightarrow c[2i] \geq b[i]$
III. $a[i] \geq b[i] \Rightarrow c[2i] \leq a[i]$
IV. $a[i] \geq b[i] \Rightarrow c[2i] \leq b[i]$

Which of the following is TRUE?

A. only I and II
B. only I and IV
C. only II and III
D. only III and IV

gate2005-it algorithms sorting normal

Answer

1.18.39 Sorting: GATE IT 2008 | Question: 43 top

https://gateoverflow.in/3353

If we use Radix Sort to sort n integers in the range $\left(n^{k/2}, n^k\right]$, for some $k > 0$ which is independent of n, the time taken would be?

A. $\Theta(n)$
B. $\Theta(kn)$
C. $\Theta(n \log n)$
D. $\Theta(n^2)$

gate2008-it algorithms sorting normal

Answer

Answers: Sorting

1.18.1 Sorting: GATE CSE 1988 | Question: 1iii top

https://gateoverflow.in/91338

✔ Answer is **LESS**.

As worst case time for Quicksort is $O(n^2)$ and worst case for heap sort is $O(n \log n)$.

17 votes -- Pavan Kumar Munnam (7.4k points)

1.18.2 Sorting: GATE CSE 1990 | Question: 3-v top

https://gateoverflow.in/84830

✔ First of all which sorting algorithm is being asked? It is not just the normal ones we see in algorithm text books but can be any sorting algorithm which uses comparisons. So, in crux we cannot rely on any specific algorithms for this question.

Now, coming to the options, we can see that Θ notation is used. We use Θ notation for a problem when

1. we have a lower bound for the solution -- that is any algorithm which solves the problem must have minimum this much

complexity (time complexity being implied)
2. there exist an algorithm which solves the problem with above minimum time complexity (worst case input being implied)

For comparison based sorting we already have known algorithms like heap sort and merge sort, which solves the problem in $O(n \log n)$ (See O) and now if we show that this is the best possible by any algorithm our answer becomes $\Theta(n \log n)$.

And it is indeed proved that for comparison based sorting minimum $\Omega(n \log n)$ comparisons are required and considering time taken being proportional to the number of comparisons, the time complexity is also $\Omega(n \log n)$. Proof of this is given here but in essence it says

- The output of sorting n elements can be any of the $n!$ permutations.
- Each comparison reduces the possibility by 2
- So, to finish all $n!$ permutations we need minimum $\lg n!$ comparisons which is $\Omega(n \lg n)$

Now, if someone asks what is the minimum number of comparisons to sort 5 elements answer is definitely $\geq \lg 5! \geq \lg 120 \geq 7$. We can only use $\geq$ here and not $=$ unless we prove that we can do it in 7 comparisons. But interestingly, this $\geq$ becomes $=$ till $n = 11$ as given in below Wikipedia link.

Ref: https://en.wikipedia.org/wiki/Comparison_sort

Answer A. $\Theta(n \log n)$

References

7 votes -- Arjun Suresh (330k points)

1.18.3 Sorting: GATE CSE 1991 | Question: 01,vii top https://gateoverflow.in/505

✔ The answer is 7.

Minimum number of comparisons $= \lceil \log(n!) \rceil = \lceil \log(5!) \rceil = \lceil \log(120) \rceil = 7.$

Reference: http://en.wikipedia.org/wiki/Comparison_sort#Number_of_comparisons_required_to_sort_a_list

References

40 votes -- Rajarshi Sarkar (27.8k points)

1.18.4 Sorting: GATE CSE 1991 | Question: 13 top https://gateoverflow.in/540

✔ Answer should be counting sort which will take $O(n + k)$ time.

See here:http://www.geeksforgeeks.org/counting-sort/

References

18 votes -- One (1.7k points)

1.18.5 Sorting: GATE CSE 1992 | Question: 02,ix top https://gateoverflow.in/559

✔ Answer is (D) Part.

Although people have provided correct answers but it seems some more explanation is required.
Let there be $\mathbf{d}$ **digits** in max input integer, **b is the base** for representing input numbers and $\mathbf{n}$ **is total numbers** then **Radix Sort takes** $\mathbf{O(d * (n + b))}$ **time**. Sorting is performed from least significant digit to most significant digit.

For example, for decimal system, b is 10. What is the value of d? If k is the maximum possible value, then d would be $O(\log_b(k))$. So overall time complexity is $O((n + b) * \log_b(k))$. Which looks more than the time complexity of comparison based sorting algorithms for a large k. Let us first limit k. Let $k \leqslant n^c$ where c is a constant. In that case, the

complexity becomes $O(n \log_b(n))$. But it still does not beat comparison based sorting algorithms.
What if we make value of b larger?. What should be the value of b to make the time complexity linear? If we **set b as n then** we will get the time complexity as $O(n)$.

In other words, we can sort an array of integers with range from 1 to n^c, If the numbers are represented in base n (or every digit takes $\log_2(n)$ bits).

Reference: http://www.geeksforgeeks.org/radix-sort/

References

41 votes -- Chhotu Ram Chauhan (12k points)

1.18.6 Sorting: GATE CSE 1995 | Question: 1.16 top https://gateoverflow.in/2603

✔ **Answer: Option C.**

The number of moves are however always $m+n$ so that we can term it as $\Theta(m+n)$. But the number of comparisons varies as per the input. In the best case, the comparisons are $\min(m,n)$ and in worst case they are $m+n-1$.

40 votes -- Gate Keeda (15.9k points)

1.18.7 Sorting: GATE CSE 1995 | Question: 12 top https://gateoverflow.in/2648

✔ 1^{st} Pass: 37 52 12 11 25 92

2^{nd} Pass: 37 12 11 25 52 92

3^{rd} Pass: 12 11 25 37 52 92

4^{th} Pass: 11 12 25 37 52 92

5^{th} Pass: 11 12 25 37 52 92

Why 5^{th} pass? In 4^{th} pass, the array is already sorted, but the algorithm does not know if it is completed. The algorithm needs one **whole** pass without **any** swap to know it is sorted.

27 votes -- Gate Keeda (15.9k points)

1.18.8 Sorting: GATE CSE 1996 | Question: 14 top https://gateoverflow.in/2766

✔

a. The smallest element is at index 1,1.
b. So we have to give an array which is partially sorted. Definition of partially sorted is given in the question.
   We will give the value of x which is less than last row & column value.
   At last, 1,1 should be deleted & x should be at its correct place.

```
i=1;j=1; a[i][j]=x;
while ((x>a[i+1][j]) || (x>a[i][j+1]))
{
      if((a[i+1][j] < x) && (a[i+1][j] <a[i][j+1]))
      {
        a[i][j]=a[i+1][j];
        i=i+1;
      }
   else
      {
        a[i][j]=a[i][j+1];
        j=j+1;
      }
}
a[i][j]=x;
```

Enter the dimension of $n \times n$ array. Give the value of n
3
Enter the array elements in partially sorted order
2 3 9
5 6 10
8 11 15

Enter the value of x

7
The final output.
3 6 9
5 7 10
8 11 15

31 votes -- Ahwan Mishra (10.2k points)

1.18.9 Sorting: GATE CSE 1996 | Question: 2.15 top https://gateoverflow.in/2744

✔ Correct Option: **C**
both are the worst cases of quick sort. (assuming pivot is either first or last element)

i. is sorted in ascending order.
ii. is sorted in descending order.

26 votes -- Gate Keeda (15.9k points)

1.18.10 Sorting: GATE CSE 1998 | Question: 1.22 top https://gateoverflow.in/1659

✔

- Selection sort : $O(n^2)$
- Merge sort : $O(n \log n)$
- Binary search : $(\log n)$
- Insertion sort : $O(n)$

Note: if we use $O(n^2)$ for Insertion sort, we will not be having any suitable choice to fill selection sort. So, we can assume that the question is asking for best case time complexities.
Correct Answer: B

19 votes -- Bhagirathi Nayak (11.7k points)

1.18.11 Sorting: GATE CSE 1999 | Question: 1.12 top https://gateoverflow.in/1465

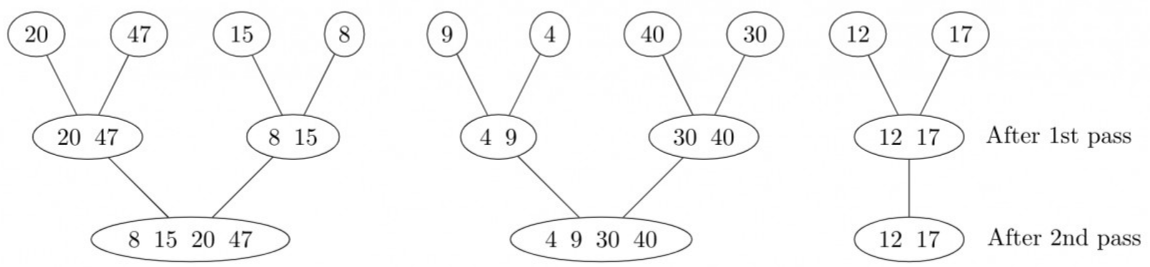
✔ Correct Option: **B**
If it maintains the relative order of occurrence of non-distinct elements.

(from definition of stable sorting)

24 votes -- Arjun Suresh (330k points)

1.18.12 Sorting: GATE CSE 1999 | Question: 1.14, ISRO2015-42 top https://gateoverflow.in/1467

✔



The answer is **B.**

94 votes -- Vikrant Singh (11.2k points)

1.18.13 Sorting: GATE CSE 1999 | Question: 8 top https://gateoverflow.in/1507

✔ This is a two dimensional array of size 4×4.
2____5____8____9
4____11___19___21
6____13___24___27
8____15___25___29

You can see the elements in each row and each column are arranged in ascending order.
Smallest element: $A[0][0] = 2$
2nd Smallest element: $min(A[0][1], A[1][0]) = min(5, 4) = 4$
3rd smallest element: Just exclude the element you got as 2^{nd} smallest(4). Here, we can compare $A[2][0], A[0][1]$ no need to compare with $A[0][2]$. S,o it depends upon from where you got 2nd element. You can draw a decision tree. If you got 2^{nd}best from $A[0][1]$ then what to do & if you get from $A[1][0]$ then what to do.

Any way, time complexity is simply **O(1)**.
The elements are in ascending order. Not in non decreasing order. Clearly they are all distinct in a particular row or column.

30 votes -- Ahwan Mishra (10.2k points)

1.18.14 Sorting: GATE CSE 2000 | Question: 17 top https://gateoverflow.in/688

✔ Since swaps are needed to be of adjacent elements only, the algorithm is actually Bubble sort.

In bubble sort, all smaller elements to right of an element are required to be swapped. So, if have ordering

$[2, 2, 2, 1, 1, 1, 1, 1, 0, 0, 0, 0]$, then we need total 47 swaps, and this will be the worst case.

So, it answers actually both parts.

38 votes -- Happy Mittal (8.2k points)

1.18.15 Sorting: GATE CSE 2001 | Question: 1.14 top https://gateoverflow.in/707

✔ Correct Option: **C**
There are two cases, when Randomized Quick Sort will result into worst case time complexity of $O(n^2)$

1. When all elements are same in the input array, Partition algorithm will divide input array in two sub-arrays, one with $n - 1$ elements and second with 0 element. There is an assumption here that, we are using the same partition algorithm without any modification.

2. If the randomised pivot selector happens to select the smallest or largest element N times in a row, we will get the worst possible performance. Though the probability of this particular case is about $\frac{2^{n-1}}{n!}$.

PS: Option D is also correct here as $n^2 = O(n!)$ though (C) is a better choice.

34 votes -- Manu Thakur (34.1k points)

In worst case, we may pick pivot elements in the increasing order (input also given in sorted order) which will result in running time of O(n^2)

Both the deterministic and randomized quicksort algorithms have the same best-case running times of O($nlogn$) and the same worst-case running times of O(n^2).The difference is that with the deterministic algorithm, a particular input can elicit that worst-case behavior. With the randomized algorithm, however, no input can always elicit the worst-case behavior. The reason it matters is that, depending on how partitioning is implemented, an input that is already sorted--or almost sorted--can elicit the worst-case behavior in deterministic quicksort.

source: Thomas Coremen

Ans. C

46 votes -- Vikrant Singh (11.2k points)

1.18.16 Sorting: GATE CSE 2003 | Question: 22 top https://gateoverflow.in/912

✔ In insertion sort, with linear search, it takes

(worst case) n comparisons for searching the right position, and n swaps to make room to place the element.

Hence for n elements, a total of $n \times (n + n)$; n for search and n for swaps.

$= \Theta(2n^2) = \Theta(n^2)$

If we replace it with binary search, it takes

(worst case) $\log n$ comparisons for searching the right position, and n swaps to make room to place the element.

Hence for n elements, a total of $n \times (\log n + n)$; n for search and n for swaps.

$= \Theta(n \times \log n + n^2) = \Theta(n^2)$

Hence, **answer is A.**

105 votes -- ryan sequeira (3k points)

1.18.17 Sorting: GATE CSE 2003 | Question: 61 top https://gateoverflow.in/949

✔ With n integers we can have ${}^nC_2 = \frac{n(n-1)}{2}$ possible pairs.

Now, since the integers are **distinct,** and selection is **random** if we take any pair, it is equally likely to be inverted or not. i.e., probability of inversion $= 0.5$.

Now, since we have ${}^nC_2$ pairs and probability of each pair being inverted $= 0.5$, expected number of inversions,

$E = \sum_{i=1}^{{}^nC_2} 0.5 = 0.5 \times \frac{n.(n-1)}{2} = \frac{n(n-1)}{4}$.

Correct Option: B.

46 votes -- Arjun Suresh (330k points)

1.18.18 Sorting: GATE CSE 2003 | Question: 62 top https://gateoverflow.in/43576

✔ As the question says, how the **Inversion** is defined .

In a permutation $a_1 \ldots a_n$, of n distinct integers, an inversion is a pair (a_i, a_j) such that $i < j$ and $a_i > a_j$.

- One important thing to see is **Difference between swap and Inversions.**
- **Swapping** is done explicitly by the programmer, hence a explicit feature whereas, **Inversion** is an implicit feature which is defined in the input .

Ex :- Take the input => $\{0, 1, 9, 8, 10, 7, 6, 11\}$

How many Inversions here : $\{9, 8\}$, $\{9, 7\}$, $\{9, 6\}$,$\{8, 7\}$, $\{8, 6\}$, $\{10, 7\}$ and $\{10, 6\}$. Hence, it is an implicit feature of the input and not any explicit operation done (Swap) .

Actual Time complexity of Insertion Sort is defined as $\Theta(N + f(N))$, where $f(N)$ is the total number of Inversions .

Ex :- Again take the input => $\{0, 6, 7, 8, 9, 10, 1\}$

Here, how many comparisons will be there to place 1 in the right place ?

- First of all, 1 is compared with 10 - Returns True as it is smaller than 10.
- Then, with 9 - again True.
- Similar, happens with $6, 7, 8$ - All return True .

Hence, There 5 comparisons were the Inversions and 1 more comparison will be there, in which outer while loop fails .

For, placing 1 in the right place **6** comparisons are there .

Hence, Total Comparisons in the Insertion sort will be :- **Total number of elements for which our while loop fails + Total number of inversions in the input**

- Total number of elements for which our while loop fails :- Suppose the input $\{0, 6, 7, 8, 9, 10, 1\}$. Here, first 0 will be kept as it is and one while loop fail comparison for each element, hence total comparisons like this :- $(N - 1) = O(N)$ comparisons.
- Total number of inversions in the input :- Best Case : 0 and in the Worst Case : $\frac{N(N-1)}{2} = O(N^2)$

Total Time complexity of insertion sort : $\Theta(N + f(N))$

It is given here that at most N inversions are there, so we get the best Time complexity :- $\Theta(N + f(N)) = \Theta(N + N) = \Theta(N)$.

Correct Answer: D

63 votes -- Kapil Phulwani (35.2k points)

1.18.19 Sorting: GATE CSE 2005 | Question: 39 top https://gateoverflow.in/784

✔ Since we have $\log n$ lists we can make a min-heap of $\log n$ elements by taking the first element from each of the $\log n$ sorted lists. Now, we start deleting the min-element from the heap and put the next element from the sorted list from which that element was added to the heap. (This identity can be done by making a structure of two values, one for the number and one for identifying the origin sorted list of that number and storing this structure in the heap). In this way each delete and the corresponding insert will take $O(\log\log n)$ time as delete in heap of size n is $O(\log n)$ and inserting an element on a heap of size n is also $O(\log n)$. (here, heap size is $\log n$). Now, we have a total of $\log n \times \frac{n}{\log n} = n$ elements. So, total time will be $O(n \log\log n)$.

Correct Answer: A

96 votes -- gatecse (62.6k points)

1.18.20 Sorting: GATE CSE 2006 | Question: 14, ISRO2011-14 top

https://gateoverflow.in/975

✔ Correct Option: **C -** Selection sort.

Because in selection the maximum swaps which can take place are $O(n)$

Because we pick up an element an find the minimum (in case of forward sorting) from the next index till the end of array and than perform the swap

Hence, $O(n)$ whereas in all other algos the swaps are greater (considering Worst-Case scenario)

27 votes -- ANKUR MAHIWAL (363 points)

1.18.21 Sorting: GATE CSE 2006 | Question: 52 top

https://gateoverflow.in/1830

✔ As we choose the pivot a median element ... so, every time we are going to have good splits guaranteed so the best case $O(n \log n)$.

Correct Answer: B

34 votes -- Bhagirathi Nayak (11.7k points)

1.18.22 Sorting: GATE CSE 2007 | Question: 14 top

https://gateoverflow.in/1212

✔ Correct Option: **A**

Irrespective of the input, merge sort always have a time complexity of $\Theta(n \log n)$.

23 votes -- Gate Keeda (15.9k points)

1.18.23 Sorting: GATE CSE 2008 | Question: 43 top

https://gateoverflow.in/455

✔ $T(n) \leq T(n/5) + T(4n/5) + n$

One part contains $n/5$ elements
and the other part contains $4n/5$ elements
$+n$ is common to all options, so we need not to worry about it.

Hence, answer is **option B.**

37 votes -- Amar Vashishth (25.2k points)

1.18.24 Sorting: GATE CSE 2009 | Question: 11 top

https://gateoverflow.in/1303

✔ The answer is **A**.

we have 1 swap in each loop and hence n swaps at max for 1 to n. Therefore the worst case number of swaps is $\Theta(n)$

25 votes -- Gate Keeda (15.9k points)

1.18.25 Sorting: GATE CSE 2009 | Question: 39 top

https://gateoverflow.in/1325

✔ **Answer is B**.

$T(n) = O(n)$ pivot selection time $+T(n/4 - 1) + T(3n/4)$

which'll give $\Theta\left(n \log n\right)$.

Pivot selection complexity is given in questions. Pivot being the $(n/4)$th smallest element, once it is found, we have two sub arrays- one of size $(n/4 - 1)$ and other of size $(3n/4)$ and for both of these we solve recursively.

40 votes -- Gate Keeda (15.9k points)

1.18.26 Sorting: GATE CSE 2012 | Question: 39 top https://gateoverflow.in/1762

✔ We are given the first character of each n strings. To sort them, it will take $O(n \log n)$ time. In the worst case we may have to do the above process 2 times, 3 times, $\ldots, n$ times. So, $n * O(n \log n) = O(n^2 \log n)$.

Correct Answer: B

81 votes -- Bhagirathi Nayak (11.7k points)

1.18.27 Sorting: GATE CSE 2013 | Question: 30 top https://gateoverflow.in/1541

✔ To sort k elements in a heap, complexity is $\Theta(k \log k)$. Lets assume there are $\frac{\log n}{\log \log n}$ elements in the heap.

$$\begin{aligned}\text{Complexity} &= \Theta\left(\frac{\log n}{\log \log n} \log\left(\frac{\log n}{\log \log n}\right)\right) \\ &= \Theta\left(\frac{\log n}{\log \log n}(\log \log n - \log \log \log n)\right) \\ &= \Theta\left(\log n - \frac{\log n \log \log \log n}{\log \log n}\right) \\ &= \Theta(\log n) \text{ (as shown below)}\end{aligned}$$

So, (**C**) is the answer.

$\log \log n > \log \log \log n$

$$\implies \frac{\log \log \log n}{\log \log n} < 1$$

$$\implies \frac{\log n \log \log \log n}{\log \log n} < \log n$$

$$\implies \Theta\left(\log n - \frac{\log n \log \log \log n}{\log \log n}\right) = \Theta(\log n)$$

157 votes -- Arjun Suresh (330k points)

1.18.28 Sorting: GATE CSE 2013 | Question: 6 top https://gateoverflow.in/1415

✔ In selection max you can do is n swaps..selecting the smallest element from all the elements and replacing it correct position so $O(n)$

Correct Answer: B

32 votes -- Bhagirathi Nayak (11.7k points)

1.18.29 Sorting: GATE CSE 2014 Set 1 | Question: 14 top https://gateoverflow.in/1780

✔ it would be $t_1 > t_2$, because the first case is the worst case of quicksort i.e. minimum number is chosen as pivot. Hence in the worst case the comparisons are high.

The splitting occurs as
$[1][2345]$
$[2][345]$
$[3][45]$
$[4][5]$

and

$[123][45]$
$[1][23][4][5]$
$[2][3]$

Number of recursive calls remain the same, but in second case the number of elements passed for the recursive call is less and hence the number of comparisons also less.

Correct Answer: C

52 votes -- Parul Agarwal (661 points)

1.18.30 Sorting: GATE CSE 2014 Set 2 | Question: 38 top https://gateoverflow.in/1997

✔ The optimal algorithm always chooses the smallest sequences for merging.

$20\ 24 - 44, 43$ comparisons
$30\ 35 - 65, 64$ comparisons
$44\ 50 - 94, 93$ comparisons
$65\ 94 - 159, 158$ comparisons

so, totally $43 + 64 + 93 + 158 = 358$ comparisons.

PS: In merge operation we do a comparison of two elements and put one element in the sorted output array. So, every comparison produces one output element. But for the last element we won't need a comparison and we simply insert it to the output array. So for n output elements we need $(n - 1)$ comparisons.

132 votes -- Arjun Suresh (330k points)

1.18.31 Sorting: GATE CSE 2014 Set 3 | Question: 14 top https://gateoverflow.in/2048

✔ Correct Option: **A** – $O(n^2)$.

When we choose the first element as the pivot, the worst case of quick sort comes if the input is sorted- either in ascending or descending order. Now, when we choose the middle element as pivot, sorted input no longer gives worst case behavior. But, there will be some permutation of the input numbers which will be giving the same worst case behavior. For example,

$1\ 2\ 3\ 4\ 5\ 6\ 7$
This array gives worst case behavior for quick sort when the first element is pivot.

$6\ 4\ 2\ 1\ 3\ 5\ 7$
This array gives the worst case behavior of $O(n^2)$ if we take middle element as the pivot- each split will be 1 element on one side and $n - 1$ elements on other side. Similarly, for any input, we can have a permutation where the behavior is like this.
So, whichever element we take as pivot it gives worst case complexity of $O(n^2)$ as long as pivot is from a fixed position (not random position as in randomized quick sort).

82 votes -- Arjun Suresh (330k points)

1.18.32 Sorting: GATE CSE 2015 Set 1 | Question: 2 top https://gateoverflow.in/8017

✔ Correct Option: **B**
Worst case for quick sort happens when 1 element is on one list and $n - 1$ elements on another list.

34 votes -- Arjun Suresh (330k points)

1.18.33 Sorting: GATE CSE 2015 Set 2 | Question: 45 top https://gateoverflow.in/8243

✔ We have to find the k^{th} smallest element.

```
if (left_end+1 > k)
```

If the above condition is true, it means we have k^{th} smallest element on the left array, whose size is left_end instead of n for the original array. (The "+1" is used because array index in C language starts from 0). So, we can do a recursive call as

- kth_smallest(a, left_end, k);

If the above condition is false, and left_end $+1 \neq k$, it means k^{th} smallest element is on the right part of the array and it will be $(k - \text{left\_end} - 1)^{th}$ element there as left_end+1 elements are gone in the left part. So, the recursive call will be

- kth_smallest(a + left_end + 1, n - left_end_1, k - left_end - 1);

Correct Option: A.

25 votes -- Arjun Suresh (330k points)

1.18.34 Sorting: GATE CSE 2015 Set 3 | Question: 27 top https://gateoverflow.in/8480

✔ The worst case time complexity of Mergesort is $k \times n \log n$ for an input of size n.

For an input of size 64, the algorithm takes $30s$. Therefore,

$$\begin{aligned} k \times 64 \log_2 64 &= 30s \\ k \times 384 &= 30s \\ \implies k &= 0.078125s \end{aligned}$$

Let the size of the problem that can be solved in 6 minutes be x. Then,

$$k \times x \log_2 x = 360s$$

From this, we get:

$$x \log_2 x = \frac{360s}{0.078125s}$$

$$\implies x = 512$$

Correct Answer: B

75 votes -- Pragy Agarwal (18.3k points)

1.18.35 Sorting: GATE CSE 2016 Set 1 | Question: 13 top https://gateoverflow.in/39660

✔ Answer is **D**.

Insertion sort: $= \Theta(n^2)$

Merge sort: $= \Theta(n \log n)$

Quick sort: $= \Theta(n^2)$

Note : here Θ is not average case since question asked worst case so Θ represent worst case only

29 votes -- Abhilash Panicker (7.6k points)

1.18.36 Sorting: GATE CSE 2016 Set 2 | Question: 13 top https://gateoverflow.in/39561

✔

I. Quicksort takes $\Theta(n^2)$ in case of already sorted input. This is true
II. This is false. If no swap happens then bubble sort can stop in single loop. $\Theta(n)$ is best case. This is false!
III. Mergesort never takes more than $\Theta(n \log n)$ This is false
IV. This is true. Insertion sort will finish in $\Theta(n)$ time in case of sorted input.

Answer **D.** I and IV.

Proof Bubble sort has best case $O(n)$:

Ref: https://en.wikipedia.org/wiki/Bubble_sort, Aduni lecture

Now quicksort taking $\mathbf{O(n \log n)}$ can happen in some cases but not all cases, so that is why I) should be considered true. Whereas Bubble sort time complexity in best case is always $\mathbf{O(n)}$. So, D is any time stronger answer than C .

Correct Answer: D

References

39 votes -- Akash Kanase (36k points)

1.18.37 Sorting: GATE CSE 2021 Set 1 | Question: 9 top https://gateoverflow.in/357443

✔ Correct Option : **C** (Insertion Sort)
Explanation:
The given array is already sorted in ascending order.
For already sorted array, different sorting algorithms behave as following :

Selection sort :
No matter how the data is arranged, there would always be comparisons and swaps made and so the time complexity for best,average and worst case is : $O(n^2)$.
In first pass, we need $n-1$ comparisons (Or n comparisons, depending on the implementation)
In second pass, we need $n-2$ comparisons (Or $n-1$ comparisons, depending on the implementation) and so on.

So, the number of comparisons required by a selection sort of n items can be computed by the formula:
$(n-1)+(n-2)+\ldots+1=(n)(n-1)/2$

Or

Number of selection sort comparisons $=(n+1)(n)/2$

Basically, number of comparisons is $\Theta(n^2)$ in all cases.

Insertion Sort :
When elements are already sorted in desired order, there are no swaps and the correct position of the element in the sorted list is the current index itself. The time complexity is : $O(n)$
Number of comparisons $=n-1$
Comparisons in total : $1+1+\ldots+1=n-1\in\Theta(n)$.

Merge Sort :
We are dividing the list into two halves, no matter if the list is sorted or not. But if the array is sorted, while merging the list, there are no swaps merging results into an array itself. Thus, the best, average and worst case time complexity is: $O(n\log n)$

Number of comparisons, in all cases, will be $O(n\log n)$

Quick Sort :
If we use the last or first element as pivot, then Quick Sort will give worst case performance if the elements are in a sorted or reverse sorted manner. So, for the given array in the question, Quick Sort will behave in worst manner and will take $O(n^2)$ comparisons.

The best and average case time complexity is : $O(n\log n)$
Number of comparisons, in best case, will be $O(n\log n)$

So, answer will be insertion sort.

6 votes -- Deepak Poonia (23.3k points)

1.18.38 Sorting: GATE IT 2005 | Question: 59 top https://gateoverflow.in/3820

✔ $a[i]\geq b[i]$

Since both a and b are sorted in the beginning, there are i elements smaller than or equal to $a[i]$ (i starts from 0), and similarly i elements smaller than or equal to $b[i]$. So, $a[i]\geq b[i]$ means there are $2i$ elements smaller than or equal to $a[i]$, and hence in the merged array $a[i]$ will come after these $2i$ elements (its index will be $>2i$). So, $c[2i]\leq a[i]$ (equality takes care of the "equal to" case which comes when array contains repeated elements).

Similarly, $a[i]\geq b[i]$ says for b that, there are not more than $2i$ elements smaller than $b[i]$ in the sorted array (i elements from b, and maximum another i elements from a). So, $b[i]\leq c[2i]$

So, II and III are correct is option (**C**).

44 votes -- Arjun Suresh (330k points)

1.18.39 Sorting: GATE IT 2008 | Question: 43 top https://gateoverflow.in/3353

✔ **Answer: C**

The complexity of Radix Sort is $O(wn)$, for n keys which are integers of word size w.

Here, $w=\log_2(n^k)=k\times\log_2(n)$

So, the complexity is $O(wn)=O(k\times\log_2(n)\times n)$, which leads to option **C**.

56 votes -- Rajarshi Sarkar (27.8k points)

1.19 Space Complexity (1) top

1.19.1 Space Complexity: GATE CSE 2005 | Question: 81a top https://gateoverflow.in/1403

```
double foo(int n)
{
    int i;
    double sum;
    if(n == 0)
    {
        return 1.0;
    }
    else
    {
        sum = 0.0;
        for(i = 0; i < n; i++)
        {
            sum += foo(i);
        }
        return sum;
    }
}
```

The space complexity of the above code is?

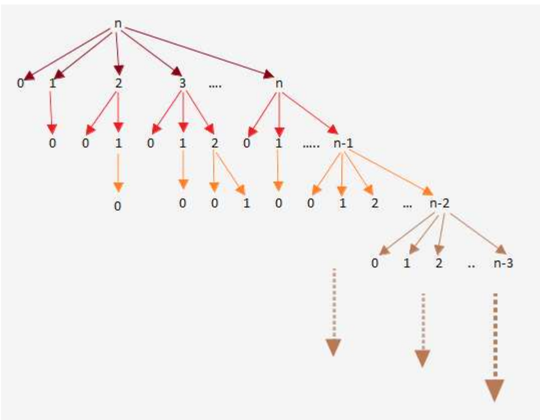
A. $O(1)$
B. $O(n)$
C. $O(n!)$
D. n^n

gate2005-cse algorithms recursion normal space-complexity

Answer

Answers: Space Complexity

1.19.1 Space Complexity: GATE CSE 2005 | Question: 81a top https://gateoverflow.in/1403



A. The code here is storing only local variables. So, the space complexity will be the recursion depth- maximum happening for the last iteration of the for loop$-foo(n-1) - foo(n-2) - \ldots - foo(0)$ all live giving space complexity $O(n)$.

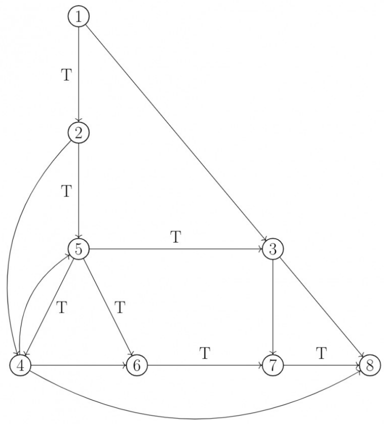
B. To store the n values we need space complexity $O(n)$. So, the space complexity won't change and will be $O(n)$.

Correct Answer: B

52 votes -- Arjun Suresh (330k points)

1.20 Spanning Tree (27) top

1.20.1 Spanning Tree: GATE CSE 1989 | Question: 4-vii top https://gateoverflow.in/88152



In the graph shown above, the depth-first spanning tree edges are marked with a $'T'$. Identify the forward, backward, and cross edges.

gate1989 descriptive algorithms graph-algorithms spanning-tree dfs

Answer

1.20.2 Spanning Tree: GATE CSE 1991 | Question: 03,vi top

https://gateoverflow.in/521

Kruskal's algorithm for finding a minimum spanning tree of a weighted graph G with n vertices and m edges has the time complexity of:

A. $O(n^2)$
B. $O(mn)$
C. $O(m+n)$
D. $O(m \log n)$
E. $O(m^2)$

gate1991 algorithms graph-algorithms spanning-tree minimum-spanning-trees time-complexity multiple-selects

Answer

1.20.3 Spanning Tree: GATE CSE 1992 | Question: 01,ix top

https://gateoverflow.in/549

Complexity of Kruskal's algorithm for finding the minimum spanning tree of an undirected graph containing n vertices and m edges if the edges are sorted is _______

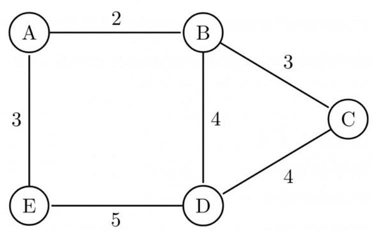
gate1992 spanning-tree algorithms time-complexity easy fill-in-the-blanks

Answer

1.20.4 Spanning Tree: GATE CSE 1995 | Question: 22 top

https://gateoverflow.in/2660

How many minimum spanning trees does the following graph have? Draw them. (Weights are assigned to edges).



gate1995 algorithms graph-algorithms spanning-tree easy descriptive

Answer

1.20.5 Spanning Tree: GATE CSE 1996 | Question: 16 top

https://gateoverflow.in/2768

A complete, undirected, weighted graph G is given on the vertex $\{0, 1, \ldots, n-1\}$ for any fixed 'n'. Draw the minimum spanning tree of G if

A. the weight of the edge (u, v) is $|u - v|$
B. the weight of the edge (u, v) is $u + v$

gate1996 algorithms graph-algorithms spanning-tree normal descriptive

Answer

1.20.6 Spanning Tree: GATE CSE 1997 | Question: 9 top https://gateoverflow.in/2269

Consider a graph whose vertices are points in the plane with integer co-ordinates (x, y) such that $1 \leq x \leq n$ and $1 \leq y \leq n$, where $n \geq 2$ is an integer. Two vertices (x_1, y_1) and (x_2, y_2) are adjacent iff $|x_1 - x_2| \leq 1$ and $|y_1 - y_2| \leq 1$. The weight of an edge $\{(x_1, y_1), (x_2, y_2)\}$ is $\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$

A. What is the weight of a minimum weight-spanning tree in this graph? Write only the answer without any explanations.
B. What is the weight of a maximum weight-spanning tree in this graph? Write only the answer without any explanations.

gate1997 algorithms spanning-tree normal descriptive

Answer

1.20.7 Spanning Tree: GATE CSE 2000 | Question: 2.18 top https://gateoverflow.in/665

Let G be an undirected connected graph with distinct edge weights. Let e_{max} be the edge with maximum weight and e_{min} the edge with minimum weight. Which of the following statements is false?

A. Every minimum spanning tree of G must contain e_{min}
B. If e_{max} is in a minimum spanning tree, then its removal must disconnect G
C. No minimum spanning tree contains e_{max}
D. G has a unique minimum spanning tree

gate2000-cse algorithms spanning-tree normal

Answer

1.20.8 Spanning Tree: GATE CSE 2001 | Question: 15 top https://gateoverflow.in/756

Consider a weighted undirected graph with vertex set
$V = \{n1, n2, n3, n4, n5, n6\}$ and edge set
$E = \{(n1, n2, 2), (n1, n3, 8), (n1, n6, 3), (n2, n4, 4), (n2, n5, 12), (n3, n4, 7), (n4, n5, 9), (n4, n6, 4)\}$.
The third value in each tuple represents the weight of the edge specified in the tuple.

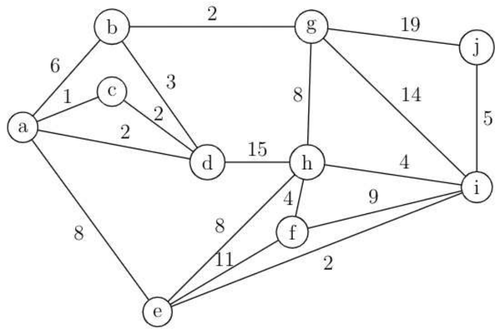
A. List the edges of a minimum spanning tree of the graph.
B. How many distinct minimum spanning trees does this graph have?
C. Is the minimum among the edge weights of a minimum spanning tree unique over all possible minimum spanning trees of a graph?
D. Is the maximum among the edge weights of a minimum spanning tree unique over all possible minimum spanning tree of a graph?

gate2001-cse algorithms spanning-tree normal descriptive

Answer

1.20.9 Spanning Tree: GATE CSE 2003 | Question: 68 top https://gateoverflow.in/955

What is the weight of a minimum spanning tree of the following graph?



A. 29
B. 31
C. 38
D. 41

gate2003-cse algorithms spanning-tree normal

Answer

1.20.10 Spanning Tree: GATE CSE 2005 | Question: 6 top https://gateoverflow.in/1348

An undirected graph G has n nodes. its adjacency matrix is given by an $n \times n$ square matrix whose (i) diagonal elements are 0's and (ii) non-diagonal elements are 1's. Which one of the following is TRUE?

A. Graph G has no minimum spanning tree (MST)
B. Graph G has unique MST of cost $n-1$
C. Graph G has multiple distinct MSTs, each of cost $n-1$
D. Graph G has multiple spanning trees of different costs

gate2005-cse algorithms spanning-tree normal

Answer

1.20.11 Spanning Tree: GATE CSE 2006 | Question: 11 top https://gateoverflow.in/890

Consider a weighted complete graph G on the vertex set $\{v_1, v_2, \dots \dots v_n\}$ such that the weight of the edge (v_i, v_j) is $2|i-j|$. The weight of a minimum spanning tree of G is:

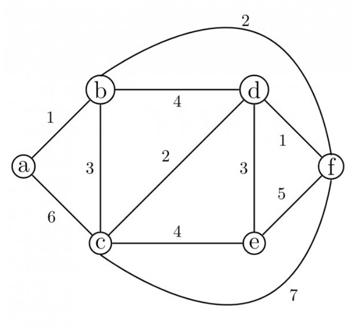
A. $n-1$
B. $2n-2$
C. $\binom{n}{2}$
D. n^2

gate2006-cse algorithms spanning-tree normal

Answer

1.20.12 Spanning Tree: GATE CSE 2006 | Question: 47 top https://gateoverflow.in/1823

Consider the following graph:



Which one of the following cannot be the sequence of edges added, **in that order**, to a minimum spanning tree using Kruskal's algorithm?

A. $(a-b),(d-f),(b-f),(d-c),(d-e)$
B. $(a-b),(d-f),(d-c),(b-f),(d-e)$
C. $(d-f),(a-b),(d-c),(b-f),(d-e)$
D. $(d-f),(a-b),(b-f),(d-e),(d-c)$

gate2006-cse algorithms graph-algorithms spanning-tree normal

Answer

1.20.13 Spanning Tree: GATE CSE 2007 | Question: 49 top

https://gateoverflow.in/1247

Let w be the minimum weight among all edge weights in an undirected connected graph. Let e be a specific edge of weight w. Which of the following is FALSE?

A. There is a minimum spanning tree containing e
B. If e is not in a minimum spanning tree T, then in the cycle formed by adding e to T, all edges have the same weight.
C. Every minimum spanning tree has an edge of weight w
D. e is present in every minimum spanning tree

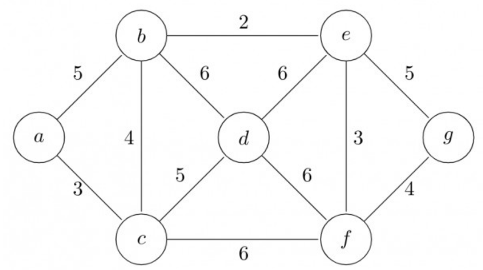
gate2007-cse algorithms spanning-tree normal

Answer

1.20.14 Spanning Tree: GATE CSE 2009 | Question: 38 top

https://gateoverflow.in/1324

Consider the following graph:



Which one of the following is NOT the sequence of edges added to the minimum spanning tree using Kruskal's algorithm?

A. (b, e) (e, f) (a, c) (b, c) (f, g) (c, d)
B. (b, e) (e, f) (a, c) (f, g) (b, c) (c, d)
C. (b, e) (a, c) (e, f) (b, c) (f, g) (c, d)
D. (b, e) (e, f) (b, c) (a, c) (f, g) (c, d)

gate2009-cse algorithms spanning-tree normal

Answer

1.20.15 Spanning Tree: GATE CSE 2010 | Question: 50 top

https://gateoverflow.in/2355

Consider a complete undirected graph with vertex set $\{0, 1, 2, 3, 4\}$. Entry W_{ij} in the matrix W below is the weight of the edge $\{i, j\}$

$$W = \begin{pmatrix} 0 & 1 & 8 & 1 & 4 \\ 1 & 0 & 12 & 4 & 9 \\ 8 & 12 & 0 & 7 & 3 \\ 1 & 4 & 7 & 0 & 2 \\ 4 & 9 & 3 & 2 & 0 \end{pmatrix}$$

What is the minimum possible weight of a spanning tree T in this graph such that vertex 0 is a leaf node in the tree T?

A. 7
B. 8
C. 9
D. 10

gate2010-cse algorithms spanning-tree normal

Answer

1.20.16 Spanning Tree: GATE CSE 2010 | Question: 51 top

https://gateoverflow.in/43328

Consider a complete undirected graph with vertex set $\{0, 1, 2, 3, 4\}$. Entry W_{ij} in the matrix W below is the weight of the edge $\{i, j\}$

$$W = \begin{pmatrix} 0 & 1 & 8 & 1 & 4 \\ 1 & 0 & 12 & 4 & 9 \\ 8 & 12 & 0 & 7 & 3 \\ 1 & 4 & 7 & 0 & 2 \\ 4 & 9 & 3 & 2 & 0 \end{pmatrix}$$

What is the minimum possible weight of a path P from vertex 1 to vertex 2 in this graph such that P contains at most 3 edges?

A. 7
B. 8
C. 9
D. 10

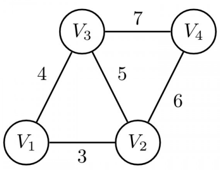
gate2010-cse normal algorithms spanning-tree

Answer

1.20.17 Spanning Tree: GATE CSE 2011 | Question: 54 top

https://gateoverflow.in/2162

An undirected graph $G(V, E)$ contains n $(n > 2)$ nodes named $v_1, v_2, \dots, v_n$. Two nodes v_i, v_j are connected if and only if $0 < | i - j | \leq 2$. Each edge (v_i, v_j) is assigned a weight $i + j$. A sample graph with $n = 4$ is shown below.



What will be the cost of the minimum spanning tree (MST) of such a graph with n nodes?

A. $\frac{1}{12}(11n^2 - 5n)$
B. $n^2 - n + 1$
C. $6n - 11$
D. $2n + 1$

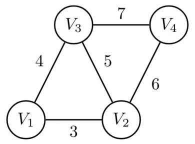
gate2011-cse algorithms graph-algorithms spanning-tree normal

Answer

1.20.18 Spanning Tree: GATE CSE 2011 | Question: 55 top

https://gateoverflow.in/43325

An undirected graph $G(V, E)$ contains n $(n > 2)$ nodes named $v_1, v_2, \ldots, v_n$. Two nodes v_i, v_j are connected if and only if $0 < |i - j| \leq 2$. Each edge (v_i, v_j) is assigned a weight $i + j$. A sample graph with $n = 4$ is shown below.



The length of the path from v_5 to v_6 in the MST of previous question with $n = 10$ is

A. 11
B. 25
C. 31
D. 41

gate2011-cse algorithms graph-algorithms spanning-tree normal

Answer

1.20.19 Spanning Tree: GATE CSE 2012 | Question: 29 top

https://gateoverflow.in/786

Let G be a weighted graph with edge weights greater than one and G' be the graph constructed by squaring the weights of edges in G. Let T and T' be the minimum spanning trees of G and G', respectively, with total weights t and t'. Which of the following statements is **TRUE**?

A. $T' = T$ with total weight $t' = t^2$
B. $T' = T$ with total weight $t' < t^2$
C. $T' \neq T$ but total weight $t' = t^2$
D. None of the above

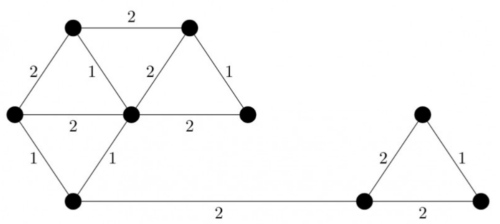
gate2012-cse algorithms spanning-tree normal marks-to-all

Answer

1.20.20 Spanning Tree: GATE CSE 2014 Set 2 | Question: 52 top

https://gateoverflow.in/2019

The number of distinct minimum spanning trees for the weighted graph below is _____



made a small mistake

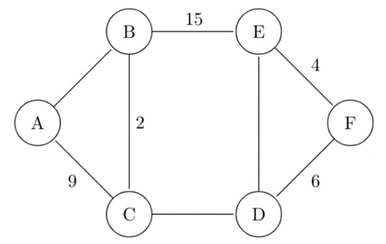
gate2014-cse-set2 algorithms spanning-tree numerical-answers normal

Answer

1.20.21 Spanning Tree: GATE CSE 2015 Set 1 | Question: 43 top

https://gateoverflow.in/8313

The graph shown below has 8 edges with distinct integer edge weights. The minimum spanning tree (**MST**) is of weight 36 and contains the edges: $\{(A, C), (B, C), (B, E), (E, F), (D, F)\}$. The edge weights of only those edges which are in the **MST** are given in the figure shown below. The minimum possible sum of weights of all 8 edges of this graph is_______________.



gate2015-cse-set1 algorithms spanning-tree normal numerical-answers

Answer

1.20.22 Spanning Tree: GATE CSE 2015 Set 3 | Question: 40 top

https://gateoverflow.in/8499

Let G be a connected undirected graph of 100 vertices and 300 edges. The weight of a minimum spanning tree of G is 500. When the weight of each edge of G is increased by five, the weight of a minimum spanning tree becomes ______.

gate2015-cse-set3 algorithms spanning-tree easy numerical-answers

Answer

1.20.23 Spanning Tree: GATE CSE 2016 Set 1 | Question: 14 top

https://gateoverflow.in/39673

Let G be a weighted connected undirected graph with distinct positive edge weights. If every edge weight is increased by the same value, then which of the following statements is/are TRUE?

- P: Minimum spanning tree of G does not change.
- Q: Shortest path between any pair of vertices does not change.

A. P only
B. Q only
C. Neither P nor Q
D. Both P and Q

V. good Q

gate2016-cse-set1 algorithms spanning-tree normal

Answer

1.20.24 Spanning Tree: GATE CSE 2016 Set 1 | Question: 39 top

https://gateoverflow.in/39725

Let G be a complete undirected graph on 4 vertices, having 6 edges with weights being $1, 2, 3, 4, 5,$ and 6. The maximum possible weight that a minimum weight spanning tree of G can have is __________

gate2016-cse-set1 algorithms spanning-tree normal numerical-answers

excellent question

Answer

1.20.25 Spanning Tree: GATE CSE 2016 Set 1 | Question: 40 top

https://gateoverflow.in/39727

$G = (V, E)$ is an undirected simple graph in which each edge has a distinct weight, and e is a particular edge of G. Which of the following statements about the minimum spanning trees $(MSTs)$ of G is/are TRUE?

I. If e is the lightest edge of some cycle in G, then every MST of G includes e.
II. If e is the heaviest edge of some cycle in G, then every MST of G excludes e.

A. I only.
B. II only.
C. Both I and II.
D. Neither I nor II.

Wonderful Q. Brain storming

gate2016-cse-set1 algorithms spanning-tree normal

Answer

A

1.20.26 Spanning Tree: GATE IT 2005 | Question: 52 top

☛ https://gateoverflow.in/3813

Let G be a weighted undirected graph and e be an edge with maximum weight in G. Suppose there is a minimum weight spanning tree in G containing the edge e. Which of the following statements is always TRUE?

A. There exists a cutset in G having all edges of maximum weight.
B. There exists a cycle in G having all edges of maximum weight.
C. Edge e cannot be contained in a cycle.
D. All edges in G have the same weight.

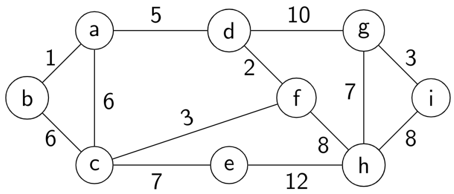
gate2005-it algorithms spanning-tree normal

Answer

1.20.27 Spanning Tree: GATE IT 2008 | Question: 45 top

☛ https://gateoverflow.in/3355

For the undirected, weighted graph given below, which of the following sequences of edges represents a correct execution of Prim's algorithm to construct a Minimum Spanning Tree?



A. (a, b), (d, f), (f, c), (g, i), (d, a), (g, h), (c, e), (f, h)
B. (c, e), (c, f), (f, d), (d, a), (a, b), (g, h), (h, f), (g, i)
C. (d, f), (f, c), (d, a), (a, b), (c, e), (f, h), (g, h), (g, i)
D. (h, g), (g, i), (h, f), (f, c), (f, d), (d, a), (a, b), (c, e)

gate2008-it algorithms graph-algorithms spanning-tree normal

Answer

Answers: Spanning Tree

1.20.1 Spanning Tree: GATE CSE 1989 | Question: 4-vii top

☛ https://gateoverflow.in/88152

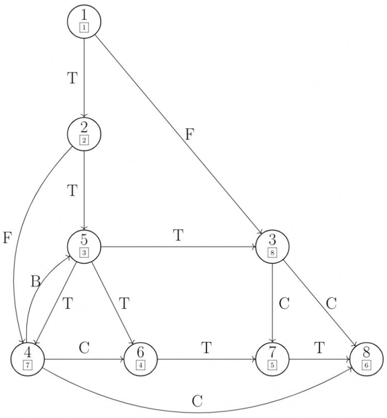
✔

1. A **tree edge (T)** is an edge in a DFS-tree.
2. A **back edge (B)** connects a vertex to an ancestor in a DFS-tree. (Includes a self-loop and this indicates the presence of a cycle)
3. A **forward edge (F)** is a non-tree edge that connects a vertex to a descendent in a DFS-tree.
4. A **cross edge (C)** is any other edge in graph G. It either connects vertices in two different DFS tree or two vertices in the same DFS tree neither of which is the ancestor of the other.

For the given question,

- Forward Edges: $1 \to 3, 2 \to 4$
- Back Edges: $4 \to 5$
- Cross Edges: $3 \to 7, 4 \to 6, 3 \to 8, 4 \to 8$

These are marked in the below graph. Below each node, its discovery time is marked. An edge (u, v) becomes a forward edge only if $d(u) < d(v)$.



PS:

- An edge (u, v) is a cross edge if and only if $d[v] < f[v] < d[u] < f[u]$. ($v$ is discovered and finished before u is discovered)
- An edge (u, v) is a back edge if and only if $d[v] < d[u] < f[u] < f[v]$. ($v$ is discovered first and before it finishes u is discovered and finished)

Reference: http://www.personal.kent.edu/~rmuhamma/Algorithms/MyAlgorithms/GraphAlgor/depthSearch.htm

References

6 votes -- Arjun Suresh (330k points)

1.20.2 Spanning Tree: GATE CSE 1991 | Question: 03,vi top https://gateoverflow.in/521

✔ Answer: B, D, E.

When Union- Find algorithm is used to detect cycle while constructing the MST time complexity is $O(m \log n)$ where m is the number of edges, and n is the number of vertices. Since $m = O\left(n^2\right)$ in a graph, options B and E are also correct as big-O specifies asymptotic upper bound only.

Reference: http://www.geeksforgeeks.org/greedy-algorithms-set-2-kruskals-minimum-spanning-tree-mst/

References

25 votes -- Rajarshi Sarkar (27.8k points)

1.20.3 Spanning Tree: GATE CSE 1992 | Question: 01,ix top https://gateoverflow.in/549

✔ If all edges are already sorted then this problem will reduced to union-find problem on a graph with E edges and V vertices.

```
for each edge (u,v) in E

    if(FIND-SET(u) != FIND-SET(v))
        UNION(u,v)
```

FIND-SET(v) and UNION(u, v) runs in $\alpha(|V|)$

where $\alpha(n)$ is inverse ackermann function i.e $\log^*(n)$

So, overall complexity becomes $O(|E|.\alpha(|V|))$

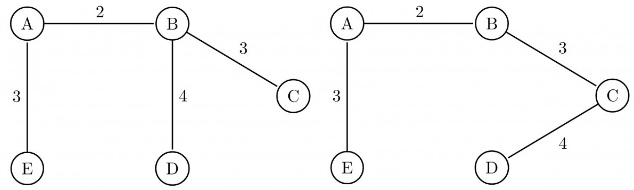
34 votes -- Vikrant Singh (11.2k points)

1.20.4 Spanning Tree: GATE CSE 1995 | Question: 22 top

https://gateoverflow.in/2660

✔ 2 only.

$\{AB, BC, AE, BD\}$ and $\{AB, BC, AE, CD\}$.



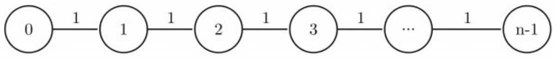
👍 28 votes

-- Gate Keeda (15.9k points)

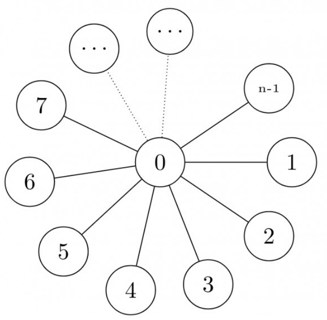
1.20.5 Spanning Tree: GATE CSE 1996 | Question: 16 top

https://gateoverflow.in/2768

✔ (A)



(B)



👍 37 votes

-- Anu (4.7k points)

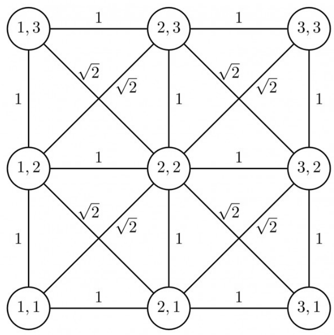
1.20.6 Spanning Tree: GATE CSE 1997 | Question: 9 top

https://gateoverflow.in/2269

✔ Consider $n = 3$, we have 9 vertices in the plane. The vertices are:

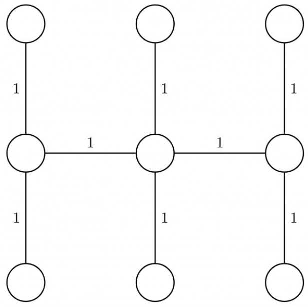
$(1, 1), \quad (1, 2), \quad (1, 3), \quad (2, 1), \quad (2, 2), \quad (2, 3), \quad (3, 1), \quad (3, 2), \quad (3, 3)$

So, the corresponding graph will be:



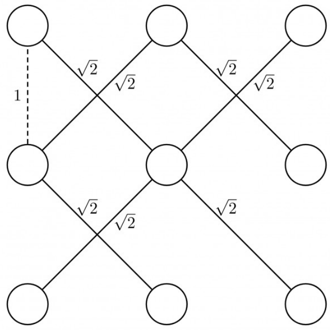
Minimum spanning tree:

(One of the possibilities)



There is no problem with a minimum spanning tree. We have n^2 vertices, so the minimum spanning tree contains $n^2 - 1$ edges of cost 1. So, the cost of the minimum spanning tree is $n^2 - 1$.

Maximum spanning tree:



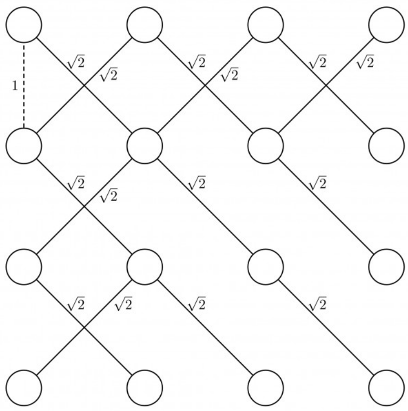
In this maximum spanning tree of $n^2 - 1$ edges , $n^2 - 2$ edges are of cost $\sqrt{2}$ and 1 edge is of cost 1.

So, the answer is $\sqrt{2}\,\left(n^2 - 2\right) + 1$.

This pattern continues for high values of n.

For $n = 4$, the maximum spanning tree is:

(one of possibilities)



For $n = 4$, 16 vertices are there and the maximum spanning tree requires 15 edges. Out of 15 edges, 14 edges are of cost $\sqrt{2}$ and 1 edge is of cost 1. Thus satisfying the formula $\sqrt{2}\,\left(n^2 - 2\right) + 1$.

The cost of Minimum spanning tree is: $n^2 - 1$.

The cost of Maximum spanning tree is: $\left(n^2 - 2\right)\sqrt{2} + 1$.

52 votes -- balaeinstein (1.2k points)

1.20.7 Spanning Tree: GATE CSE 2000 | Question: 2.18 top

https://gateoverflow.in/665

✔ **C** the case should be written as "may or may not", to be true.

D will always be true as per the question saying that the graph has distinct weights.

Correct Answer: C

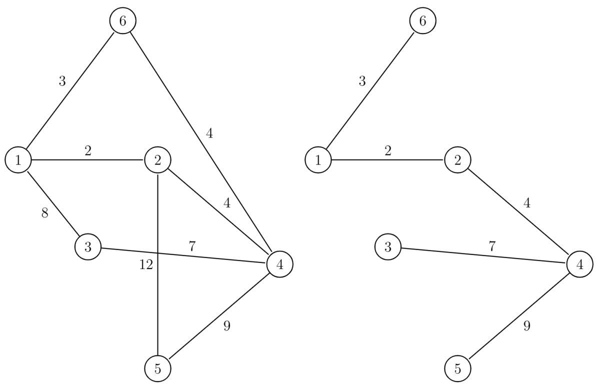
31 votes -- Gate Keeda (15.9k points)

1.20.8 Spanning Tree: GATE CSE 2001 | Question: 15 top

https://gateoverflow.in/756

✔

A. Edges with weights: $2, 3, 4, 7, 9$



Minimum Spanning Tree

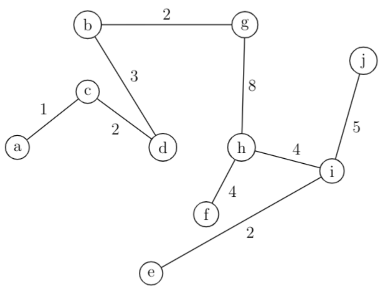
B. Number of distinct minimum spanning tree: $2 (2^{\text{nd}}$ with a different edge of weight $4)$
C. Yes
D. Yes

24 votes -- jayendra (6.7k points)

1.20.9 Spanning Tree: GATE CSE 2003 | Question: 68 top

https://gateoverflow.in/955

✔ Apply Prim's algorithm, start from A as shown in figure below.



Add all the weights in the given figure which will be equal to 31.

Correct Answer: B

16 votes -- Monanshi Jain (7k points)

1.20.10 Spanning Tree: GATE CSE 2005 | Question: 6 top

https://gateoverflow.in/1348

✔ Graph G has multiple distinct MSTs, each of cost $n-1$

From the given data given graph is a complete graph with all edge weights 1. A MST will contain $n-1$ edges . Hence weight of MST is $n-1$.

The graph will have multiple MST. In fact all spanning trees of the given graph wll be MSTs also since all edge weights are equal.

32 votes -- Sankaranarayanan P.N (8.5k points)

1.20.11 Spanning Tree: GATE CSE 2006 | Question: 11 top

https://gateoverflow.in/890

✔ $2(n-1)$ the spanning tree will traverse adjacent edges since they contain the least weight.

33 votes -- anshu (2.7k points)

1.20.12 Spanning Tree: GATE CSE 2006 | Question: 47 top https://gateoverflow.in/1823

✔ In Kruskal's algo the edges are added in non decreasing order of their weight. But in Option D edge $d-e$ with weight 3 is added before edge $d-c$ with weight 2. Hence, option **D** is wrong option.

Correct Answer: D

23 votes -- Sankaranarayanan P.N (8.5k points)

1.20.13 Spanning Tree: GATE CSE 2007 | Question: 49 top https://gateoverflow.in/1247

✔ **D** is the false statement.

A minimum spanning tree must have the edge with the smallest weight (In Kruskal's algorithm we start from the smallest weight edge). So, **C** is TRUE.

If e is not part of a minimum spanning tree, then all edges which are part of a cycle with e, must have weight $\leq e$, as otherwise we can interchange that edge with e and get another minimum spanning tree of lower weight. So, **B** and **A** are also TRUE.

D is false because, suppose a cycle is there with all edges having the same minimum weight w. Now, any one of them can be avoided in any minimum spanning tree.

31 votes -- Arjun Suresh (330k points)

1.20.14 Spanning Tree: GATE CSE 2009 | Question: 38 top https://gateoverflow.in/1324

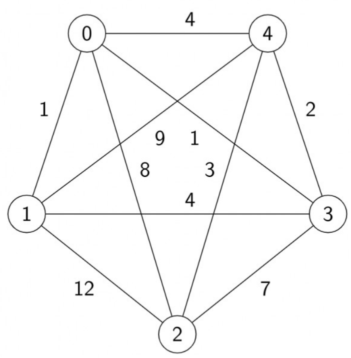
✔ In **Option D** b-c with weight, 4 is added before a-c with weight 3 is added. In Kruskal's algorithm, edges should be added in non-decreasing order of weight.

So, **Option D** may be correct.

21 votes -- Sankaranarayanan P.N (8.5k points)

1.20.15 Spanning Tree: GATE CSE 2010 | Question: 50 top https://gateoverflow.in/2355

✔

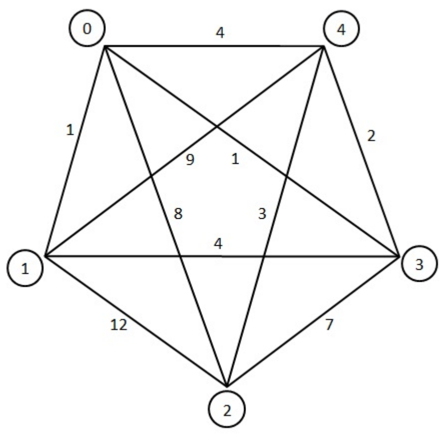


Answer is (**D**) 10. The edges of the spanning tree are: $0-1, 1-3, 3-4, 4-2$. Total Weight $=10$

38 votes -- Ashis Kumar Sahoo (699 points)

1.20.16 Spanning Tree: GATE CSE 2010 | Question: 51 top https://gateoverflow.in/43328

✔



Answer is (**B**) 8. The possible path is: $1-0, 0-4, 4-2$.

👍 25 votes -- Ashis Kumar Sahoo (699 points)

1.20.17 Spanning Tree: GATE CSE 2011 | Question: 54 top ☛ https://gateoverflow.in/2162

✔ **Q 54**. Answer is **B.**

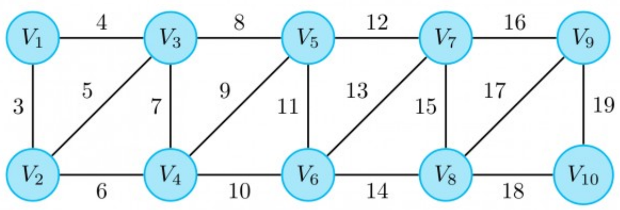
We observe a pattern in the weight of MST being formed

For n=3 $(1+2+3)+(1)$
For n=4 $(1+2+3+4)+(1+2)$
For n=5 $(1+2+3+4+5)+(1+2+3)$
These can be obtained by drawing graphs for these graphs.
$\therefore$ Total weight of MST is $\sum_{i=1}^{n} i + \sum_{i=1}^{n-2} i = n^2 - n + 1$

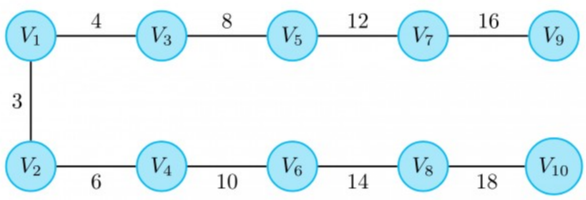
👍 39 votes -- Sona Praneeth Akula (3.4k points)

1.20.18 Spanning Tree: GATE CSE 2011 | Question: 55 top ☛ https://gateoverflow.in/43325

✔



Above is the graph.
Below is the MST.



Length of the path from v_5 to $v_6 = 8+4+3+6+10 = 31$ **(Answer)**

Correct Answer: C

👍 65 votes -- Ahwan Mishra (10.2k points)

1.20.19 Spanning Tree: GATE CSE 2012 | Question: 29 top ☛ https://gateoverflow.in/786

✔ When the edge weights are squared the minimum spanning tree won't change.

$t' < t^2$, because sum of squares is always less than the square of the sums except for a single element case.

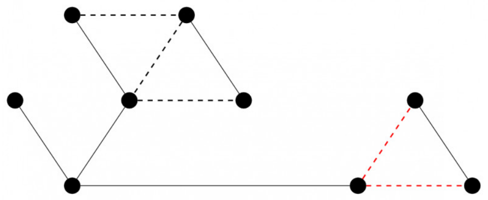
Hence, B is the general answer and A is also true for a single edge graph. Hence, in GATE 2012, marks were given to all.

61 votes -- gatecse (62.6k points)

1.20.20 Spanning Tree: GATE CSE 2014 Set 2 | Question: 52 top https://gateoverflow.in/2019

✔ 6 is the answer.

$2 \times 3 = 6$ possibilities



61 votes -- Arjun Suresh (330k points)

1.20.21 Spanning Tree: GATE CSE 2015 Set 1 | Question: 43 top https://gateoverflow.in/8313

✔ Consider the cycle ABC. AC and AB are part of minimum spanning tree. So, AB should be greater than $max(AC, BC)$ (greater and not equal as edge weights are given to be distinct), as otherwise we could add AB to the minimum spanning tree and removed the greater of AC, BC and we could have got another minimum spanning tree. So, $AB > 9$.

Similarly, for the cycle $DEF, ED > 6$.

And for the cycle $BCDE, CD > 15$.

So, minimum possible sum of these will be $10 + 7 + 16 = 33$. Adding the weight of spanning tree, we get the total sum of edge weights

$= 33 + 36 = 69$

90 votes -- Arjun Suresh (330k points)

1.20.22 Spanning Tree: GATE CSE 2015 Set 3 | Question: 40 top https://gateoverflow.in/8499

✔ First find no of edges in mst.
Mst has $n - 1$ edges where n is no of vertices. $100 - 1 = 99$ edges
Each 99 edges in mst increases by 5 so weight in mst increased $99 * 5 = 495$
Now total weight of mst $= 500 + 495 = 995$

63 votes -- Anoop Sonkar (4.1k points)

1.20.23 Spanning Tree: GATE CSE 2016 Set 1 | Question: 14 top https://gateoverflow.in/39673

✔ Statement P is true.

For statement Q consider a simple graph with 3 nodes.

	A	B	C
A	0	1	100
B	1	0	2
C	100	2	0

Shortest path from A to C is A-B-C $= 1 + 2 = 3$

Now if the value of each edge is increased by 100,

	A	B	C
A	0	101	200
B	101	0	102
C	200	102	0

The shortest path from A to C is A-C $= 200$, (A-B-C $= 101 + 102 = 203$)

Hence, option **A is correct.**

👍 85 votes -- ryan sequeira (3k points)

1.20.24 Spanning Tree: GATE CSE 2016 Set 1 | Question: 39 top https://gateoverflow.in/39725

✔ Many people here have not understood the question itself. Consider a complete graph of 4 vertices. We have a total of 6 edges of given weights but we do not have the exact graph. Many different graphs are possible each having a different structure. Consider these 2 graphs, both of them are different. We do not know the exact structure of the graph, so what the question wants is to find the MST of all such structures and out of these tell the weight of the MST having maximum weight. The point about the MST of a graph with unique edge weights is valid for a given structure of the graph. With the same set of edge weights more than 1 graph is possible and all of them can have different MSTs.

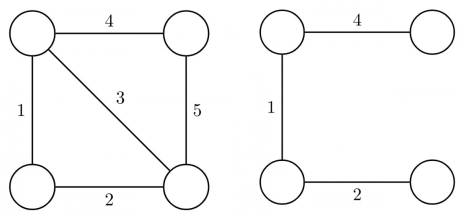
My solution: Draw a complete graph of 4 vertices. Sort given edges y weight in increasing order. Just like Kruskal's algorithm sort the edges by weight. MST of graph with 4 vertices and 6 edges will have 3 edges. Now in any case we will have to include edges with weights 1 and 2 as they are minimum and Kruskal's algorithm includes minimum weight edge if it does not form a cycle. We can not have a cycle with 2 edges. In Kruskals algorithm, an edge will be rejected if it forms a cycle with the edges already selected. To increase the weight of our MST we will try to reject the edge with weight $\mathbf{3}$. This can be done by forming a cycle. The graph in pic1 shows this case. This implies, the total weight of this graph will be $1 + 2 + 4 = \mathbf{7}$.

👍 116 votes -- air1 (2.4k points)

1.20.25 Spanning Tree: GATE CSE 2016 Set 1 | Question: 40 top https://gateoverflow.in/39727

✔ **Statement 1:-** False by [Cut Property of MST]

See counter example :- Here in below Graph G in $($cycle $3, 4, 5)$ 3 is the lightest edge and still it is not included in MST.



G MST

Statement 2: True by[Cycle property of MST] : (in above Graph G $1 - 2 - 3$ is a cycle and 3 is the heaviest edge) If heaviest edge is in cycle then we will always exclude that because cycle is there means, we can have other choice of low cost edges. (If edge weights are not distinct even the heaviest edge can be part of the MST and here in question distinct edge weight is clearly mentioned)

So, option **B** is answer.

Must visit Link: http://www.cs.princeton.edu/courses/archive/spr07/cos226/lectures/mst.pdf

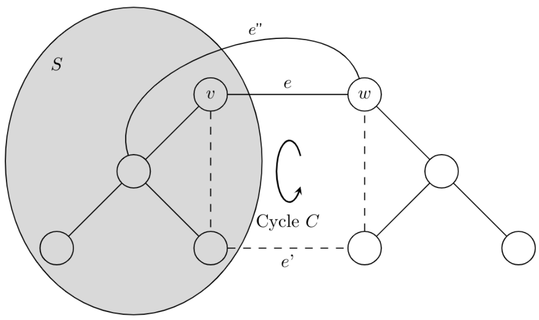
References

👍 105 votes -- Rajesh Pradhan (18.9k points)

$\mathbf{1}$. For an edge to be **INCLUDED** in the MST, it must satisfy the **CUT PROPERTY** which states that

- Assume that all edge costs are **distinct**. Let S be any subset of nodes that is neither empty nor equal to all of V, and let edge $e = (v, w)$ be the minimum cost edge with one end in S and the other in $V - S$. Then every minimum spanning tree contains the edge e.

It is given that e is the lightest edge of some cycle in G, but it may not be the lightest edge connecting S and $V - S$. See the below image for clarification.



Here, **e is the lightest edge of cycle C.** There are three edges $e, e^{'}, e''$ connecting S and $V - S$.

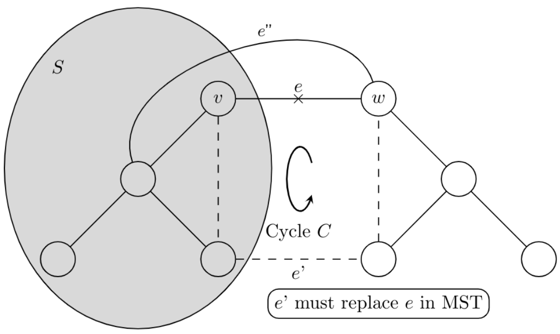
e is obviously minimal weighted than e' since e' is part of the cycle C, but e **MAY NOT** be minimal weighted than e''.

So, every MST of G **MAY NOT** include e.

2. For an edge to be **EXCLUDED** from the MST, it must satisfy the **CYCLE PROPERTY** which states that

- Assume that all edge costs are distinct. Let C be any cycle in G, and let edge $e = (v, w)$ be the most expensive edge belonging to C. Then e does not belong to the minimum spanning tree of G.

It is given that e is the **heaviest edge** of some cycle C.



But, there will always be an edge in the cycle, say e', which is lighter than e connecting S and $V - S$.

So, every MST of G **MUST** exclude e.

ANSWER is B.

Courtesy: *Algorithm Design by Jon Kleinberg and Eva Tardos*

52 votes -- Tanaya Pradhan (701 points)

1.20.26 Spanning Tree: GATE IT 2005 | Question: 52 top https://gateoverflow.in/3813

✔ Option A is correct.

Questions says the MST of graph G contain an edge e which is a maximum weight edge in G. Need to choose the answer which is always true to follow the above constraint.

Case 1:

Option B says tht if edge e ia in MST then for sure there is a cycle having all edges of maximum weight. But it **is not true always because when there is only n-1 edges(but no cycle) in graph then also maximum edge has to be taken for MST**.

Case 2:

Option C says otherwise. That if e is in MST then it cannot be in any cycle that is wrong as if there is a cycle with all maximum edges then also e will be in MST

Option D says all edges should be of same weight same explanation if there are $n - 1$ distinct edges(but no cycle) in G then have to take all edges including maximum weight edge.

And at last option A **says if e is in MST then for sure there is a cut-set (minimum edge set whose removal disconnects the graph) in G having all edges of maximum weight. And it is true.**

Because then only we maximum weight edges has to be taken in MST.

For eg. If there are $n-1$ edges (but no cycle) then if edge e is not taken in the MST then MST will not be connected.

31 votes -- khush tak (5.9k points)

1.20.27 Spanning Tree: GATE IT 2008 | Question: 45 top

https://gateoverflow.in/3355

✔ Prim's algorithm starts with from any vertex and expands the MST by adding one vertex in each step which is close to the Intermediate MST(made till previous step).

Therefore, correct answer would be (**C**).

(A): (d, f) is chosen but neither d nor f vertices are part of the previous MST(MST made till previous step).

(B): (g, h) is chosen but neither g or h vertices are part of the previous MST(MST made till previous step).

(D): (f, c) is chosen but at that point (f, d) is close to the intermediate MST.

26 votes -- suraj (4.8k points)

1.21 Time Complexity (29) top

1.21.1 Time Complexity: GATE CSE 1988 | Question: 6i top

https://gateoverflow.in/94363

Given below is the sketch of a program that represents the path in a two-person game tree by the sequence of active procedure calls at any time. The program assumes that the payoffs are real number in a limited range; that the constant INF is larger than any positive payoff and its negation is smaller than any negative payoff and that there is a function “payoff” and that computes the payoff for any board that is a leaf. The type “boardtype” has been suitably declared to represent board positions. It is player-1’s move if mode = MAX and player- 2’s move if mode=MIN. The type modetype =(MAX, MIN). The functions “min” and “max” find the minimum and maximum of two real numbers.

```
function search(B: boardtype; mode: modetype): real;
    var
        C:boardtype; {a child of board B}
        value:real;
    begin
        if B is a leaf then
            return (payoff(B))
        else
            begin
            if mode = MAX then value :=-INF
            else
                value:INF;
            for each child C of board B do
            if mode = MAX then
                value:=max (value, search (C, MIN))
            else
                value:=min(value, search(C, MAX))
            return(value)
            end
    end; (search)
```

Comment on the working principle of the above program. Suggest a possible mechanism for reducing the amount of search.

gate1988 normal descriptive algorithms time-complexity

Answer

1.21.2 Time Complexity: GATE CSE 1989 | Question: 2-iii top

https://gateoverflow.in/87080

Match the pairs in the following:

(A)	$O(\log n)$	(p)	Heapsort
(B)	$O(n)$	(q)	Depth-first search
(C)	$O(n \log n)$	(r)	Binary search
(D)	$O(n^2)$	(s)	Selection of the k^{th} smallest element in a set of n elements

gate1989 match-the-following algorithms time-complexity

Answer

1.21.3 Time Complexity: GATE CSE 1993 | Question: 8.7 top https://gateoverflow.in/2305

$\sum_{1 \leq k \leq n} O(n)$, where $O(n)$ stands for order n is:

A. $O(n)$
B. $O(n^2)$
C. $O(n^3)$
D. $O(3n^2)$
E. $O(1.5n^2)$

gate1993 algorithms time-complexity easy

Answer

1.21.4 Time Complexity: GATE CSE 1999 | Question: 1.13 top https://gateoverflow.in/1466

Suppose we want to arrange the n numbers stored in any array such that all negative values occur before all positive ones. Minimum number of exchanges required in the worst case is

A. $n-1$
B. n
C. $n+1$
D. None of the above

gate1999 algorithms time-complexity normal

Answer

1.21.5 Time Complexity: GATE CSE 1999 | Question: 1.16 top https://gateoverflow.in/1469

If n is a power of 2, then the minimum number of multiplications needed to compute a^n is

A. $\log_2 n$
B. $\sqrt{n}$
C. $n-1$
D. n

gate1999 algorithms time-complexity normal

Answer

1.21.6 Time Complexity: GATE CSE 1999 | Question: 11a top https://gateoverflow.in/1510

Consider the following algorithms. Assume, procedure A and procedure B take $O(1)$ and $O(1/n)$ unit of time respectively. Derive the time complexity of the algorithm in O-notation.

```
algorithm what (n)
begin
    if n = 1 then call A
    else
        begin
            what (n-1);
            call B(n)
        end
```

```
end.
```

gate1999 algorithms time-complexity normal descriptive

Answer

1.21.7 Time Complexity: GATE CSE 2000 | Question: 1.15 top

https://gateoverflow.in/638

Let S be a sorted array of n integers. Let $T(n)$ denote the time taken for the most efficient algorithm to determined if there are two elements with sum less than 1000 in S. Which of the following statement is true?

A. $T(n)$ is $O(1)$
B. $n \leq T(n) \leq n \log_2 n$
C. $n \log_2 n \leq T(n) < \frac{n}{2}$
D. $T(n) = \binom{n}{2}$

gate2000-cse easy algorithms time-complexity

Answer

1.21.8 Time Complexity: GATE CSE 2003 | Question: 66 top

https://gateoverflow.in/258

The cube root of a natural number n is defined as the largest natural number m such that $(m^3 \leq n)$. The complexity of computing the cube root of n (n is represented by binary notation) is

A. $O(n)$ but not $O(n^{0.5})$
B. $O(n^{0.5})$ but not $O((\log n)^k)$ for any constant $k > 0$
C. $O((\log n)^k)$ for some constant $k > 0$, but not $O((\log \log n)^m)$ for any constant $m > 0$
D. $O((\log \log n)^k)$ for some constant $k > 0.5$, but not $O((\log \log n)^{0.5})$

gate2003-cse algorithms time-complexity normal

Answer

1.21.9 Time Complexity: GATE CSE 2004 | Question: 39 top

https://gateoverflow.in/1036

Two matrices M_1 and M_2 are to be stored in arrays A and B respectively. Each array can be stored either in row-major or column-major order in contiguous memory locations. The time complexity of an algorithm to compute $M_1 \times M_2$ will be

A. best if A is in row-major, and B is in column-major order
B. best if both are in row-major order
C. best if both are in column-major order
D. independent of the storage scheme

gate2004-cse algorithms time-complexity easy

Answer

1.21.10 Time Complexity: GATE CSE 2004 | Question: 82 top

https://gateoverflow.in/1076

Let $A[1, \dots, n]$ be an array storing a bit (1 or 0) at each location, and $f(m)$ is a function whose time complexity is $\Theta(m)$. Consider the following program fragment written in a C like language:

```
counter = 0;
for (i=1; i<=n; i++)
{
    if a[i] == 1) counter++;
    else {f (counter); counter = 0;}
}
```

The complexity of this program fragment is

A. $\Omega(n^2)$
B. $\Omega(n \log n)$ and $O(n^2)$
C. $\Theta(n)$

D. $o(n)$

gate2004-cse algorithms time-complexity normal

Answer

1.21.11 Time Complexity: GATE CSE 2006 | Question: 15 top https://gateoverflow.in/976

Consider the following C-program fragment in which i, j and n are integer variables.

```
for( i = n, j = 0; i > 0; i /= 2, j +=i );
```

Let $val(j)$ denote the value stored in the variable j after termination of the for loop. Which one of the following is true?

A. $val(j) = \Theta(\log n)$
B. $val(j) = \Theta(\sqrt{n})$
C. $val(j) = \Theta(n)$
D. $val(j) = \Theta(n \log n)$

gate2006-cse algorithms normal time-complexity

Answer

1.21.12 Time Complexity: GATE CSE 2007 | Question: 15,ISRO2016-26 top https://gateoverflow.in/56129

Consider the following segment of C-code:

```
int j, n;
j = 1;
while (j <= n)
    j = j * 2;
```

The number of comparisons made in the execution of the loop for any $n > 0$ is:

A. $\lceil \log_2 n \rceil + 1$

B. n

C. $\lceil \log_2 n \rceil$

D. $\lfloor \log_2 n \rfloor + 1$

gate2007-cse algorithms time-complexity normal isro2016

Answer

1.21.13 Time Complexity: GATE CSE 2007 | Question: 44 top https://gateoverflow.in/1242

In the following C function, let $n \geq m$.

```
int gcd(n,m)   {
    if (n%m == 0) return m;
    n = n%m;
    return gcd(m,n);
}
```

How many recursive calls are made by this function?

A. $\Theta(\log_2 n)$

B. $\Omega(n)$

C. $\Theta(\log_2 \log_2 n)$

D. $\Theta(\sqrt{n})$

gate2007-cse algorithms time-complexity normal

Answer

1.21.14 Time Complexity: GATE CSE 2007 | Question: 45 top https://gateoverflow.in/1243

What is the time complexity of the following recursive function?

```
int  DoSomething (int n) {
    if (n <= 2)
        return 1;
    else
        return (DoSomething (floor (sqrt(n))) + n);
}
```

A. $\Theta(n^2)$
B. $\Theta(n \log_2 n)$
C. $\Theta(\log_2 n)$
D. $\Theta(\log_2 \log_2 n)$

gate2007-cse algorithms time-complexity normal

Answer

1.21.15 Time Complexity: GATE CSE 2007 | Question: 50 top

https://gateoverflow.in/1248

An array of n numbers is given, where n is an even number. The maximum as well as the minimum of these n numbers needs to be determined. Which of the following is **TRUE** about the number of comparisons needed?

A. At least $2n - c$ comparisons, for some constant c are needed.
B. At most $1.5n - 2$ comparisons are needed.
C. At least $n \log_2 n$ comparisons are needed
D. None of the above

gate2007-cse algorithms time-complexity easy

Answer

1.21.16 Time Complexity: GATE CSE 2007 | Question: 51 top

https://gateoverflow.in/1249

Consider the following C program segment:

```
int IsPrime (n)
    {
        int i, n;
        for (i=2; i<=sqrt(n);i++)
            if(n%i == 0)
                {printf("Not Prime \n"); return 0;}
        return 1;
    }
```

Let $T(n)$ denote number of times the for loop is executed by the program on input n. Which of the following is TRUE?

A. $T(n) = O(\sqrt{n})$ and $T(n) = \Omega(\sqrt{n})$
B. $T(n) = O(\sqrt{n})$ and $T(n) = \Omega(1)$
C. $T(n) = O(n)$ and $T(n) = \Omega(\sqrt{n})$
D. None of the above

gate2007-cse algorithms time-complexity normal

Answer

1.21.17 Time Complexity: GATE CSE 2008 | Question: 40 top

https://gateoverflow.in/452

The minimum number of comparisons required to determine if an integer appears more than $\frac{n}{2}$ times in a sorted array of n integers is

A. $\Theta(n)$
B. $\Theta(\log n)$
C. $\Theta(\log^* n)$
D. $\Theta(1)$

gate2008-cse normal algorithms time-complexity

Answer

1.21.18 Time Complexity: GATE CSE 2008 | Question: 47 top https://gateoverflow.in/459

We have a binary heap on n elements and wish to insert n more elements (not necessarily one after another) into this heap. The total time required for this is

A. $\Theta(\log n)$
B. $\Theta(n)$
C. $\Theta(n \log n)$
D. $\Theta(n^2)$

gate2008-cse algorithms time-complexity normal

Answer

1.21.19 Time Complexity: GATE CSE 2008 | Question: 74 top https://gateoverflow.in/495

Consider the following C functions:

```
int f1 (int n)
{
    if(n == 0 || n == 1)
        return n;
    else
        return (2 * f1(n-1) + 3 * f1(n-2));
}
int f2(int n)
{
    int i;
    int X[N], Y[N], Z[N];
    X[0] = Y[0] = Z[0] = 0;
    X[1] = 1; Y[1] = 2; Z[1] = 3;
    for(i = 2; i <= n; i++){
        X[i] = Y[i-1] + Z[i-2];
        Y[i] = 2 * X[i];
        Z[i] = 3 * X[i];
    }
    return X[n];
}
```

The running time of $f1(n)$ and $f2(n)$ are

A. $\Theta(n)$ and $\Theta(n)$
B. $\Theta(2^n)$ and $\Theta(n)$
C. $\Theta(n)$ and $\Theta(2^n)$
D. $\Theta(2^n)$ and $\Theta(2^n)$

gate2008-cse algorithms time-complexity normal

Answer

1.21.20 Time Complexity: GATE CSE 2008 | Question: 75 top https://gateoverflow.in/43489

Consider the following C functions:

```
int f1 (int n)
{
    if(n == 0 || n == 1)
        return n;
    else
        return (2 * f1(n-1) + 3 * f1(n-2));
}
int f2(int n)
{
    int i;
    int X[N], Y[N], Z[N];
    X[0] = Y[0] = Z[0] = 0;
    X[1] = 1; Y[1] = 2; Z[1] = 3;
    for(i = 2; i <= n; i++){
        X[i] = Y[i-1] + Z[i-2];
        Y[i] = 2 * X[i];
```

```
        Z[i] = 3 * X[i];
    }
    return X[n];
}
```

$f1(8)$ and $f2(8)$ return the values

A. 1661 and 1640
B. 59 and 59
C. 1640 and 1640
D. 1640 and 1661

gate2008-cse normal algorithms time-complexity

Answer

1.21.21 Time Complexity: GATE CSE 2010 | Question: 12 top

https://gateoverflow.in/2185

Two alternative packages A and B are available for processing a database having 10^k records. Package A requires $0.0001n^2$ time units and package B requires $10n \log_{10} n$ time units to process n records. What is the smallest value of k for which package B will be preferred over A?

A. 12
B. 10
C. 6
D. 5

gate2010-cse algorithms time-complexity easy

Answer

1.21.22 Time Complexity: GATE CSE 2014 Set 1 | Question: 42 top

https://gateoverflow.in/1920

Consider the following pseudo code. What is the total number of multiplications to be performed?

```
D = 2
for i = 1 to n do
    for j = i to n do
        for k = j + 1 to n do
            D = D * 3
```

A. Half of the product of the 3 consecutive integers.
B. One-third of the product of the 3 consecutive integers.
C. One-sixth of the product of the 3 consecutive integers.
D. None of the above.

gate2014-cse-set1 algorithms time-complexity normal

Answer

1.21.23 Time Complexity: GATE CSE 2015 Set 1 | Question: 40 top

https://gateoverflow.in/8299

An algorithm performs $(\log N)^{\frac{1}{2}}$ find operations , N insert operations, $(\log N)^{\frac{1}{2}}$ delete operations, and $(\log N)^{\frac{1}{2}}$ decrease-key operations on a set of data items with keys drawn from a linearly ordered set . For a delete operation, a pointer is provided to the record that must be deleted . For the decrease-key operation, a pointer is provided to the record that has its key decreased. Which one of the following data structures is the most suited for the algorithm to use, if the goal is to achieve the best total asymptotic complexity considering all the operations?

A. Unsorted array
B. Min - heap
C. Sorted array
D. Sorted doubly linked list

gate2015-cse-set1 algorithms data-structures normal time-complexity

Answer

1.21.24 Time Complexity: GATE CSE 2015 Set 2 | Question: 22 top

https://gateoverflow.in/8113

An unordered list contains n distinct elements. The number of comparisons to find an element in this list that is neither maximum nor minimum is

A. $\Theta(n \log n)$
B. $\Theta(n)$
C. $\Theta(\log n)$
D. $\Theta(1)$

gate2015-cse-set2 algorithms time-complexity easy

Answer

1.21.25 Time Complexity: GATE CSE 2017 Set 2 | Question: 03 top

https://gateoverflow.in/118156

Match the algorithms with their time complexities:

Algorithms	Time Complexity
P. Tower of Hanoi with n disks	i. $\Theta(n^2)$
Q. Binary Search given n sorted numbers	ii. $\Theta(n \log n)$
R. Heap sort given n numbers at the worst case	iii. $\Theta(2^n)$
S. Addition of two $n \times n$ matrices	iv. $\Theta(\log n)$

A. $P \to (iii) \quad Q \to (iv) \quad r \to (i) \quad S \to (ii)$
B. $P \to (iv) \quad Q \to (iii) \quad r \to (i) \quad S \to (ii)$
C. $P \to (iii) \quad Q \to (iv) \quad r \to (ii) \quad S \to (i)$
D. $P \to (iv) \quad Q \to (iii) \quad r \to (ii) \quad S \to (i)$

gate2017-cse-set2 algorithms time-complexity

Answer

1.21.26 Time Complexity: GATE CSE 2017 Set 2 | Question: 38 top

https://gateoverflow.in/118283

Consider the following C function

```
int fun(int n) {
    int i, j;
    for(i=1; i<=n; i++) {
        for (j=1; j<n; j+=i) {
            printf("%d %d", i, j);
        }
    }
}
```

Time complexity of fun in terms of Θ notation is

A. $\Theta(n\sqrt{n})$
B. $\Theta(n^2)$
C. $\Theta(n \, \log n)$
D. $\Theta(n^2 \log n)$

gate2017-cse-set2 algorithms time-complexity

Answer

1.21.27 Time Complexity: GATE CSE 2019 | Question: 37 top

https://gateoverflow.in/302811

There are n unsorted arrays: $A_1, A_2, \dots, A_n$. Assume that n is odd.Each of $A_1, A_2, \dots, A_n$ contains n distinct elements. There are no common elements between any two arrays. The worst-case time complexity of computing the median of the medians of $A_1, A_2, \dots, A_n$ is

A. $O(n)$
B. $O(n \, \log \, n)$
C. $O(n^2)$
D. $\Omega(n^2 \log n)$

gate2019-cse algorithms time-complexity

Answer

1.21.28 Time Complexity: GATE IT 2007 | Question: 17 top

https://gateoverflow.in/3450

Exponentiation is a heavily used operation in public key cryptography. Which of the following options is the tightest upper bound on the number of multiplications required to compute $b^n \bmod m, 0 \leq b, n \leq m$?

A. $O(\log n)$
B. $O(\sqrt{n})$
C. $O\left(\frac{n}{\log n}\right)$
D. $O(n)$

gate2007-it algorithms time-complexity normal

Answer

1.21.29 Time Complexity: GATE IT 2007 | Question: 81 top

https://gateoverflow.in/3533

Let $P_1, P_2, \ldots, P_n$ be n points in the xy-plane such that no three of them are collinear. For every pair of points P_i and P_j, let L_{ij} be the line passing through them. Let L_{ab} be the line with the steepest gradient among all $n(n-1)/2$ lines.

The time complexity of the best algorithm for finding P_a and P_b is

A. $\Theta(n)$
B. $\Theta(n \log n)$
C. $\Theta\left(n \log^2 n\right)$
D. $\Theta\left(n^2\right)$

gate2007-it algorithms time-complexity normal

Answer

Answers: Time Complexity

1.21.1 Time Complexity: GATE CSE 1988 | Question: 6i top

https://gateoverflow.in/94363

Here, the game is started by giving the first move to a player.

Let's assume Player 1 takes the first move. At initial step, his payoff will be the maximum payoff achieved by Player 2 for all possible child boards as shown by the following statement:

```
value:=max (value, search (C, MIN))
```

Similarly, Player 2 will get the minimum payoff achieved by Player 1 for all the possible child boards.

The problem with the above algorithm is that the same path in the tree will be evaluated repeatedly for different parent node (here node is the board). The amount of search can be significantly reduced if the output of each recursive call is saved and possibly reused if required again, instead of doing a recursive call for each child board.

6 votes

-- Arjun Suresh (330k points)

1.21.2 Time Complexity: GATE CSE 1989 | Question: 2-iii top

https://gateoverflow.in/87080

✔

(A)	$O(\log n)$	(r)	Binary search
(B)	$O(n)$	(s)	Selection of the k^{th} smallest element in a set of n elements (Worst case)
(C)	$O(n \log n)$	(p)	Heap sort
(D)	$O(n^2)$	(q)	Depth-first search (It will be $O(n+m)$ if the graph is given in the form of adjacency list. But if the graph is given the form of adjacency matrix then the complexity is $O(n \times n)$, as we have to traverse through the whole row until we find an edge)

PS: k^{th} smallest element can be found in $O(n)$ time using partition algorithm.

$T(n) = T(n/2) + n$ (For finding k^{th} smallest element) if the graph is given in the form of adjacency list and by Masters Theorem it will be done in $O(n)$.

20 votes -- Prajwal Bhat (7.6k points)

1.21.3 Time Complexity: GATE CSE 1993 | Question: 8.7 top

https://gateoverflow.in/2305

✔ This is N added itself N times. So it is N^2 . Even if you consider as sum of $O(1) + O(2) + \ldots + O(n-1) + O(N)$. it will add up to N^2

So, the answer is:

(A) $O(N)$ this is false.

(B, C, D, E) All of this is true. We have N^2 here, so all options apart from (A) are correct.

In fact $B = D = E$ this three options are same. and N^3 is always the upper bound of N^2. So $O(N^3)$ is also true.

PS: $\sum(K = 1 \text{ to } n)O(K)$ is never equal to n functions. It is always equal to one single function.

It can be written as $c.1 + c.2 + c.3$ and so on which results in $O(N^2)$ (source Cormen)

40 votes -- Akash Kanase (36k points)

1.21.4 Time Complexity: GATE CSE 1999 | Question: 1.13 top

https://gateoverflow.in/1466

✔ Answer is (**D**) None of these.

We just require $n/2$ swaps in the worst case. The algorithm is as given below:

Find positive number from left side and negative number from right side and do exchange. Since, at least one of them must be less than or equal to $n/2$, there cannot be more than $n/2$ exchanges. An implementation is given below:

```
#include<stdio.h>
int main() {

       int i, n, pi, ni, count = 0;
       int a[1000];
       printf("Enter size of array: ");
       scanf("%d",&n);
       printf("Enter numbers of array\n");
       for(i=0; i<n; i++)
       {
               scanf("%d",&a[i]);
       }
       ni = n-1;
       /*Making ni point to the rightmost negative number*/
       while(a[ni] >= 0)
               ni--;
       pi = 0;
       /*Making pi point to the leftmost positive number*/
       while(a[pi] < 0)
               pi++;
       /*Looping till either negative or positive numbers exhaust*/
       while(ni > pi)
       {
               /*Swapping a[ni] and a[pi]*/
```

```
                int temp = a[pi];
                a[pi] = a[ni];
                a[ni] = temp;
                /*Moving ni leftwards to the next negative number*/
                while(a[ni] >= 0)
                        ni--;
                /*Moving pi rightwards to the next positive number*/
                while(a[pi] < 0)
                        pi++;
        }
        for(i=0; i<n; i++)
        {
                printf("%d ", a[i]);
        }

}
```

75 votes -- Arjun Suresh (330k points)

1.21.5 Time Complexity: GATE CSE 1999 | Question: 1.16 top

https://gateoverflow.in/1469

✔ Correct Option: **A** ($\log_2 n$)

$a^n = (a^2)^{\frac{n}{2}}$.

One multiplication and recurrence on $\frac{n}{2}$. So, we get the recurrence relation for the number of multiplications as

$T(n) = T(n/2) + 1.$

This gives $T(n) = \log_2 n$

For n = 8, we can do

$b = a \times a$

$b = b \times b$

$b = b \times b$ and we get $b = a^8$

38 votes -- Arjun Suresh (330k points)

1.21.6 Time Complexity: GATE CSE 1999 | Question: 11a top

https://gateoverflow.in/1510

✔ The recurrence relation for time complexity is

$T(n) = T(n-1) + \frac{1}{n} + c$ ($O(1/n)$ replaced with $1/n$ and so our answer will also be in O only and not Θ)

$$\begin{aligned} T(n) &= T(n-2) + \tfrac{1}{n-1} + \tfrac{1}{n} + 2c \\ &= T(1) + \tfrac{1}{2} + \tfrac{1}{3} + \dots + \tfrac{1}{n} + (n-1)c \\ &= A(1) + \tfrac{1}{2} + \tfrac{1}{3} \dots + \tfrac{1}{n} + nc \\ &= 1 + \tfrac{1}{2} + \tfrac{1}{3} \dots + \tfrac{1}{n} + nc \\ &= \log n + nc \end{aligned}$$

(Sum of the first n terms in harmonic series is $\Theta(\log n)$)

So, our time complexity will be $O(n)$.

45 votes -- Arjun Suresh (330k points)

1.21.7 Time Complexity: GATE CSE 2000 | Question: 1.15 top

https://gateoverflow.in/638

✔ **Answer**: **Option A**. Because array is always sorted just check the 1st two elements.

49 votes -- anshu (2.7k points)

1.21.8 Time Complexity: GATE CSE 2003 | Question: 66 top

https://gateoverflow.in/258

✔ We can simply do a binary search in the array of natural numbers from $1..n$ and check if the cube of the number matches n (i.e., check if $a[i] * a[i] * a[i] == n$). This check takes $O(\log n)$ time and in the worst case we need to do the search $O(\log n)$ times. So, in this way we can find the cube root in $O(\log^2 n)$. So, options (A) and (B) are wrong.

Now, a number is represented in binary using $\log n$ bit. Since each bit is important in finding the cube root, any cube root finding algorithm must examine each bit at least once. This ensures that complexity of cube root finding algorithm cannot be lower than $\log n$. (It must be $\Omega(\log n)$). So, (D) is also false and (C) is the correct answer.

93 votes -- gatecse (62.6k points)

1.21.9 Time Complexity: GATE CSE 2004 | Question: 39 top

https://gateoverflow.in/1036

✔ **D** is correct

Here time complexity is asked, for each access of array element it will be constant,

So the time complexity will not depend upon storage. If at all program execution time is asked option a is true.

50 votes -- Anurag Semwal (6.7k points)

1.21.10 Time Complexity: GATE CSE 2004 | Question: 82 top

https://gateoverflow.in/1076

✔ The key part in the code is "$\text{counter} = 0$" in the else part as we can see below.

Lets take the best case. This happens when $a[i] = 1$ for all i, and then the loop executes with time complexity $\Theta(1)$ for each iteration and hence overall time complexity of $\Theta(n)$ and we can say time complexity of the code fragment is $\Omega(n)$ and hence options A and B are false.

Now, consider the worst case. This happens when $a[i] = 0$ or when else part is executed. Here, the time complexity of each iteration will be $\Theta(\text{counter})$ and after each else, counter is reset to 0. Let k iterations go to the else part during the worst case. Then the worst case time complexity will be $\Theta(x_1) + \Theta(x_2) + \cdots + \Theta(x_k) + \Theta(n-k)$, where x_i is the value of the counter when, $A[i] = 0$ and $f(\text{counter})$ is called. But due to $\text{counter} = 0$ after each call to $f()$,

we have, totally n iterations, in which first
k iterations are performed by else part and remaining
$n - k$ iterations are performed by if part

$\therefore$ if $\text{counter} = 0$, it leads to $\Theta(1) \implies$ this will repeated k times (one time for each $xi = 0$)

. So, $\Theta(x_1) + \Theta(x_2) + \cdots + \Theta(x_k) + \Theta(n-k) = k.\Theta(1) + \Theta(n-k) = \Theta(k) + \Theta(n-k) = \Theta(n)$

Since the time complexity is $\Omega(n)$ and $\Theta(n)$ we can say it is $\Theta(n)$ - Option (C). (Option D is false because the small o needs the growth rate to be **STRICTLY lower** and not equal to or lower as the case for big O)

If $\text{counter} = 0$ was not there in else part, then time complexity would be $\Omega(n)$ and $O(n^2)$ as in worst case we can have equal number of 0's and 1's in array a giving time complexity $\Theta(1) + \Theta(2) + \cdots + \Theta(n/2) + \Theta(n/2)$ would give $O(n^2)$.

Correct Answer: C.

71 votes -- Arjun Suresh (330k points)

1.21.11 Time Complexity: GATE CSE 2006 | Question: 15 top

https://gateoverflow.in/976

✔ Answer will be $\Theta(n)$

$$j = n/2 + n/4 + n/8 + \ldots + 1$$

$$= n\left[1/2^1 + 1/2^2 + 1/2^3 + \ldots + 1/2^{\lg n}\right]$$

(Sum of first n terms of GP is $\left[a.\frac{1-r^n}{1-r}\right]$, where a is the first term, r is the common ratio < 1, and n is the number of terms)

$$= n\left[1/2\frac{1-(1/2)^{\lg n}}{1-1/2}\right]$$

$$= n\left[\frac{n-1}{n}\right]$$

$= n - 1 = \Theta(n)$

50 votes -- (points)

1.21.12 Time Complexity: GATE CSE 2007 | Question: 15,ISRO2016-26 top https://gateoverflow.in/56129

✔

n	No. of comparisons	$\lfloor \log_2 n \rfloor + 1$
1	$2\ (j = 1, 2)$	1
2	$3\ (j = 1, 2, 4)$	2
3	$3\ (j = 1, 2, 4)$	2
4	$4\ (j = 1, 2, 4, 8)$	3
5	$4\ (j = 1, 2, 4, 8)$	3

We have to count those comparisons which happens during the execution of the loop and so the exit comparison must also be a part. So, the correct answer should be $\lfloor \log_2 n \rfloor + 2$.

Since this is not in the choices we can assume that the question setter excluded the exit comparison and so the answer should be $\lfloor \log_2 n \rfloor + 1$.

Option D.

70 votes -- Vikrant Singh (11.2k points)

1.21.13 Time Complexity: GATE CSE 2007 | Question: 44 top https://gateoverflow.in/1242

✔ Worst case will arise when both n and m are consecutive Fibonacci numbers.

$\gcd(F_n, F_{n-1}) = \gcd(F_{n-1}, F_{n-2}) = \cdots = \gcd(F_1, F_0) = 1$

and n^{th} Fibonacci number is 1.618^n, where 1.618 is the Golden ratio.

So, to find $\gcd(n, m)$, number of recursive calls will be $\Theta(\log n)$.

Correct Answer: A

References

48 votes -- Vikrant Singh (11.2k points)

1.21.14 Time Complexity: GATE CSE 2007 | Question: 45 top https://gateoverflow.in/1243

✔ We are asked the time complexity which will be the number of recursive calls in the function as in each call we perform a constant no. of operations and a recursive call. The recurrence relation for this is (considering constant time "c" as 1)

$$T(n) = T(\sqrt{n}) + 1$$
$$= T\left(n^{1/4}\right) + 2$$
$$= T\left(n^{1/8}\right) + 3$$

Going like this we will eventually reach $T(3)$ or $T(2)$. For asymptotic case this doesn't matter and we can assume we reach $T(2)$ and in next step reach $T(1)$. So, all we want to know is how many steps it takes to reach $T(1)$ which will be 1+no. of steps to reach $T(2)$.

From the recurrence relation we know that $T(2)$ happens when $n^{\left(\frac{1}{2^k}\right)} = 2$.

Taking log and equating,

$$\frac{1}{2^k} \log n = 1$$
$$\implies 2^k = \log n$$
$$\implies k = \log \log n.$$

So, $T(1)$ happens in $\log \log n + 1$ calls, but for asymptotic complexity we can write as $\Theta\left(\log \log n\right)$

Alternatively,

Substituting values

$T(1) = 1$
$T(2) = 1$
$T(3) = T(1) + 1 = 2$
$\vdots$
$T(8) = T(2) + 1 = 2$
$T(9) = T(3) + 1 = 3$
$\vdots$

$$T\left(\left(\left(2^2\right)^2\right)^2\right) = T\left(\left(2^2\right)^2\right) + 1$$
$$= T(2^2) + 2$$
$$= T(2) + 3 = 1 + 3 = 4,$$

$\log\log n = 3$ as $n = 256$.

$$T\left(\left(\left(\left(\left(2^2\right)^2\right)^2\right)^2\right)^2\right) = 6,$$

$\log\log n = 5$ as $n = 65536 \times 65536 = 2^{32}$

$$T\left(2^{(2^{10})}\right) = T\left(2^{512}\right) + 1$$
$$= T(2^{256}) + 2$$
$$= T(2^{128}) + 3$$
$$= T(2^{64}) + 4$$
$$= T(2^{32}) + 5$$
$$= T(2^{16}) + 6$$
$$= T(2^{8}) + 7$$
$$= T(2^{4}) + 8$$
$$= T(2^{2}) + 9$$
$$= T(2) + 10 = 11,$$

$\log\log n = 10$

So, answer is **D.**

http://stackoverflow.com/questions/16472012/what-would-cause-an-algorithm-to-have-olog-log-n-complexity

References

70 votes

-- Arjun Suresh (330k points)

1.21.15 Time Complexity: GATE CSE 2007 | Question: 50 top

https://gateoverflow.in/1248

✔ An easier way to find it is by using **Tournament Method Technique** -

1. To find the smallest element in the array will take $\mathbf{n-1}$ **comparisions.**
2. To find the largest element -
    a. After the first round of Tournament , there will be exactly $\mathbf{n/2}$ **numbers** that will loose the round.
    b. So the biggest looser (the largest number) should be among these 50 loosers.To find the largest number will take $\mathbf{n/2-1}$.

Total Comparisons $= \mathbf{(n-1)+(n/2-1)=1.5n-2}$.

Correct Answer: B

137 votes

-- Harsh181996 (3k points)

1.21.16 Time Complexity: GATE CSE 2007 | Question: 51 top

https://gateoverflow.in/1249

✔ Answer is **option B.**

Worst Case : $T(n) = \mathcal{O}(\sqrt{n})$

Best Case : When n is an even number body of for loop is executed only 1 time (due to "$return$ 0" inside if) which is irrespective of n. $\therefore T(n) = \Omega(1)$

41 votes -- Gate Keeda (15.9k points)

1.21.17 Time Complexity: GATE CSE 2008 | Question: 40 top https://gateoverflow.in/452

✔ Answer is **option B.**

whenever there exists an element which is present in the array : more than $\frac{n}{2}$ times, then definitely it will be present at the middle index position; in addition to that it will also be present at anyone of the neighbourhood indices namely $i-1$ and $i+1$
No matter how we push that stream of **More than** $\frac{n}{2}$ times of elements of same value around the Sorted Array, it is bound to be present at the middle index + atleast anyone of its neighbourhood

once we got the element which should have occurred more that $n/2$ times we count its total occurrences in $\mathcal{O}(\log n)$ time.

References

57 votes -- Amar Vashishth (25.2k points)

1.21.18 Time Complexity: GATE CSE 2008 | Question: 47 top https://gateoverflow.in/459

✔ An insert operation on a binary heap takes $\mathcal{O}(\log n)$ time, but an alternative approach we can use. which requires us to insert n elements in heap without any computation i.e. in constant time. after which we can apply Heapify operation(this operation creates heap in linear time) on the array of those element and Hence obtain a Heap in $\mathcal{O}(n)$ time.

Here "not necessarily one after another" should mean that we can insert n elements at once and not necesaairly have to wait for first insert to be completed before doing second.

Correct Answer: B

57 votes -- Amar Vashishth (25.2k points)

1.21.19 Time Complexity: GATE CSE 2008 | Question: 74 top https://gateoverflow.in/495

✔ Time complexity of $f1$ is given by
$T(n) = T(n-1) + T(n-2)$, (multiplication by 2 and 3 won't affect complexity as it is a constant time operation)
$T(0) = T(1) = 1$

The solution to this (fibonacci series) is given by Golden ratio. https://en.wikipedia.org/wiki/Golden_ratio which is $O(2^n)$. (Using theta in question must be a mistake)

Time complexity of $f2$ is $\Theta(n)$ as here all recursive calls are avoided by saving the results in an array (dynamic programming).

So, answer is (**B**).

References

44 votes -- Arjun Suresh (330k points)

1.21.20 Time Complexity: GATE CSE 2008 | Question: 75 top https://gateoverflow.in/43489

✔ Both $f1$ and $f2$ are calculating the same function in recursive and iterative fashion respectively.

So. lets solve the recurrence relation.

$F(n) = 2F(n-1) + 3F(n-2)$

Its characteristic equation will be
$r^2 - 2r - 3 = 0$
$\implies (r-3)(r+1) = 0$
$\implies r = 3, -1$

Now, we have two roots. So the equation will be

$a_n = C_1(-1^n) + C_2(3^n) \qquad \rightarrow (1)$

Now from the function $f1$,

$F(0) = 0$ and $F(1) = 1$

So, $C_1 + C_2 = 0$
$\quad -C_1 + 3C_2 = 1$

$\implies C_1 = -1/4$ and $C_2 = 1/4$

After putting the values in (i) the equation will become

$a_n = (-1/4)(-1^n) + (1/4)(3^n)$

Putting $n = 8$ it will become
$F(8) = a_8 = 1640.$

Or we can do it manually

$f1(2) = 2f1(1) + 3f1(0) = 2$
$f1(3) = 2f1(2) + 3f1(1) = 7$
$f1(4) = 20$
$f1(5) = 61$
$f1(6) = 182$
$f1(7) = 547$
$f1(8) = 1640 = f2(8)$

13 votes -- Puja Mishra (6.3k points)

1.21.21 Time Complexity: GATE CSE 2010 | Question: 12 top https://gateoverflow.in/2185

✔ $10n \log_{10} n \leq 0.0001n^2$

$\implies 10 \times 10^k \log_{10} 10^k \leq 0.0001(10^k)^2$

$\implies 10^{k+1}k \leq 0.0001 \times 10^{2k}$

$\implies k \leq 10^{2k-k-1-4}$

$\implies k \leq 10^{k-5}$

Trying the values, 5 doesn't satisfy this but 6 satisfies.

Correct Answer: C

90 votes -- Arjun Suresh (330k points)

1.21.22 Time Complexity: GATE CSE 2014 Set 1 | Question: 42 top https://gateoverflow.in/1920

✔ Total number of multiplications

$= \sum_{i=1}^{n} \sum_{j=i}^{n} \sum_{k=j+1}^{n} 1$
$= \sum_{i=1}^{n} \sum_{j=i}^{n} (n-j)$
$= \sum_{i=1}^{n} (n-i) + (n-(i+1)) + \ldots + (n-n)$
$= \frac{1}{2} \sum_{i=1}^{n} (n-i)(n-i+1)$
$= \frac{1}{2} \sum_{i=1}^{n} (n^2 + n + i^2 - (2n+1)i)$
$= \frac{1}{2} (n^3 + n^2 + \sum_{i=1}^{n} i^2 - (2n+1). \sum_{i=1}^{n} i)$
$= \frac{1}{2} \left(n^3 + n^2 + \frac{n.(n+1).(2n+1)}{6} - (2n+1). \frac{n.(n+1)}{2}\right)$

$= \frac{1}{2}\left(n^3 + n^2 - \frac{n.(n+1).(2n+1)}{3}\right)$
$= \frac{n}{6}(3n(n+1) - (n+1).(2n+1))$
$= \frac{n.(n+1)}{6}(n-1)$
$= \frac{(n-1)n(n+1)}{6}$

Therefore, correct answer would be (**C**).

81 votes -- suraj (4.8k points)

1.21.23 Time Complexity: GATE CSE 2015 Set 1 | Question: 40 top https://gateoverflow.in/8299

	$(\log \mathbf{N})^{\frac{1}{2}}$ find	$\mathbf{N}$ insert	$(\log \mathbf{N})^{\frac{1}{2}}$ delete	$(\log \mathbf{N})^{\frac{1}{2}}$ decrease-key
Unsorted Array	$O(N(\log N)^{\frac{1}{2}})$	$O(N)$	$O(\log N)^{\frac{1}{2}})$	$O(\log N)^{\frac{1}{2}})$
Min-heap	$O(N(\log N)^{\frac{1}{2}})$	$O(N \log N)$	$O(\log N)^{\frac{3}{2}})$	$O((\log N)^{\frac{3}{2}})$
Sorted Array	$O((\log N)^{\frac{3}{2}})$	$O(N^2)$	$O(N(\log N)^{\frac{1}{2}})$	$O(N(\log N)^{\frac{1}{2}})$
Sorted doubly linked-list	$O(N(\log N)^{\frac{1}{2}})$	$O(N^2)$	$O((\log N)^{\frac{1}{2}})$	$O(N(\log N)^{\frac{1}{2}})$

So, Unsorted array is the answer. The operations given can be performed in any order. So, for Min-heap we cannot do the usual BuildHeap method. Delete in unsorted array is $O(1)$ as we can just swap the deleted element with the last element in the array and delete the last element. For sorted-doubly linked-list we cannot do binary search as this would require another array to maintain the pointers to the nodes.

So, Unsorted array is the answer.

The operations given can be performed in any order. So, for Min-heap we cannot do the usual BuildHeap method.

Delete in unsorted array is $O(1)$ as we can just swap the deleted element with the last element in the array and delete the last element.

For sorted-doubly linked-list we cannot do binary search as this would require another array to maintain the pointers to the nodes.

Correct Answer: A

109 votes -- Arjun Suresh (330k points)

1.21.24 Time Complexity: GATE CSE 2015 Set 2 | Question: 22 top https://gateoverflow.in/8113

Correct Option: **D** – $\Theta(1)$

Because all elements are distinct, select any three numbers and output 2nd largest from them.

96 votes -- Vikrant Singh (11.2k points)

1.21.25 Time Complexity: GATE CSE 2017 Set 2 | Question: 03 top https://gateoverflow.in/118156

According to the recurrence relation of the time complexity of Tower of Hanoi, $T(n) = 2T(n-1) + 1,$ we get it is $\Theta(2^n)$

Now, heap sort worst case time complexity is $\Theta(n \log n)$

Binary Search to search an element given n sorted numbers is $\Theta(\log n)$

Addition of two $n \times n$ matrices is $\Theta(n^2)$

So, **C** is correct answer here.

28 votes -- Aboveallplayer (12.5k points)

1.21.26 Time Complexity: GATE CSE 2017 Set 2 | Question: 38 top https://gateoverflow.in/118283

Inner for loop is dependent on i, so for each i we have to check no of times inner loop operating..

It ll be something like

$\frac{n-1}{1}+\frac{n-1}{2}+\frac{n-1}{3}+\ldots\ldots\ldots\ldots+\frac{n-1}{n-1}+1$

$\frac{n}{1}+\frac{n}{2}+\frac{n}{3}+\ldots\ldots+\frac{n}{n-1}-\log(n-1)$

$n\{\frac{1}{1}+\frac{1}{2}+\frac{1}{3}+\ldots\ldots+\frac{1}{n-1}\}-\log(n-1)$

$n\log(n-1)-\log(n-1)$

$n\log(n-1)$

$n\log n$

Correct Answer: C

41 votes -- 2018 (5.5k points)

1.21.27 Time Complexity: GATE CSE 2019 | Question: 37 top

https://gateoverflow.in/302811

Given that all lists are unsorted !

therefore we can't apply Binary search,

one way to find median is sorting the list, it takes $\Theta(nlogn)$, But with out sorting we can find median in $O(n)$.

For one list it takes $O(n)$, then for n-lists it takes $O(n^2)$.

So, now median of every list in our hand !

note that these medians are also not sorted !

Therefore make all these medians as one list, then with in $O(n)$ time we can find the median of medians.

$TC = O(n^2) + O(n) = O(n^2)$.

43 votes -- Shaik Masthan (50.4k points)

Here is the pseudo code for finding median in linear time. For proof ref this.

```
FindMedian(A,k){
    if (A has 10 or fewer elements){
        sort A
        return A[k-1]
    }
    partition A into subsets S[i] of five elements each
       (there will be n/5 subsets total).

    for (i = 1 to n/5)
       x[i] = FindMedian(S[i],3)
    M = FindMedian({x[i]}, n/10)

    partition A into A1<M, A2=M, A3>M
    if (k <= length(A1))
        return FindMedian(A1,k)
    else if (k > length(A1)+length(A2))
        return FindMedian(A3,k-length(A1)-length(A2))
    else return M
}
```

1. For each row find median M_i using FindMedian algorithm. $\forall\ i\ M_i \in M$
2. Find median of M

Time complexity:

1. Total n elements in each row so finding median of a row will take $O(n)$ time. Total n row so total time $n * (O(n)) = O(n^2)$
2. $|M| = n$ so finding median of M will take $O(n)$ time.

Total time $= O(n^2) + O(n) = O(n^2)$

Answer is (C)

References

18 votes -- Digvijay (44.9k points)

1.21.28 Time Complexity: GATE IT 2007 | Question: 17 top https://gateoverflow.in/3450

✔ Correct Option: **A**

We need to divide
n **recursively and compute like following:**

$C_1 = b^{\frac{n}{2}} \times b^{\frac{n}{2}}$. In this, we need to calculate $b^{\frac{n}{2}}$ only once.

$C_2 = b^{\frac{n}{4}} \times b^{\frac{n}{4}}$

$\vdots$

$$C_k = b^2 \times b^2 \qquad \begin{cases} k = \log n \end{cases}$$

Recurrence relation: $T(n) = T\left(\frac{n}{2}\right) + O(1)$

$T(n) = O(\log n)$

48 votes -- Sandeep_Uniyal (6.5k points)

1.21.29 Time Complexity: GATE IT 2007 | Question: 81 top https://gateoverflow.in/3533

✔ Answer: **B**

Gradient $= y_2 - y_1/x_2 - x_1$

For gradient to be maximum $x_2 - x_1$ should be minimum. So, sort the points (in $\Theta(n \log n)$ time) according to x coordinate and find the minimum difference between them (in $\Theta(n)$ time).

Best complexity: $\Theta(n \log n + n)$ which leads to B.

https://www.careercup.com/question?id=4787065684230144

https://stackoverflow.com/questions/8222108/max-slope-from-set-of-points

References

39 votes -- Rajarshi Sarkar (27.8k points)

Answer Keys

1.1.1	N/A	1.1.2	N/A	1.1.3	B	1.1.4	C	1.1.5	12
1.1.6	29	1.1.7	A;C	1.2.1	N/A	1.2.2	N/A	1.2.3	A
1.2.4	B	1.2.5	C	1.2.6	C	1.2.7	C	1.3.1	A;B
1.3.2	B	1.3.3	X	1.3.4	D	1.3.5	B	1.3.6	A

1.3.7	C	1.3.8	C	1.3.9	D	1.3.10	A	1.3.11	C
1.3.12	C	1.3.13	D	1.3.14	B	1.3.15	D	1.3.16	D
1.3.17	A	1.4.1	B	1.5.1	B	1.5.2	C	1.5.3	C
1.5.4	B	1.5.5	B	1.5.6	A	1.5.7	C	1.5.8	34
1.5.9	150	1.5.10	C	1.5.11	1500	1.5.12	C	1.6.1	B
1.6.2	N/A	1.6.3	N/A	1.6.4	A	1.6.5	N/A	1.6.6	B
1.6.7	D	1.6.8	B	1.6.9	B	1.6.10	D	1.6.11	A
1.6.12	A	1.6.13	A	1.6.14	D	1.6.15	D	1.6.16	D
1.6.17	D	1.6.18	C	1.6.19	B	1.6.20	D	1.6.21	C
1.6.22	C	1.6.23	6	1.6.24	B	1.6.25	A	1.6.26	109
1.6.27	A	1.6.28	99	1.6.29	A;B	1.6.30	929 : 929	1.6.31	D
1.6.32	A	1.6.33	C	1.6.34	D	1.6.35	B	1.6.36	C
1.7.1	C	1.7.2	C	1.7.3	D	1.7.4	C	1.7.5	C
1.7.6	B	1.7.7	19	1.7.8	D	1.7.9	31	1.7.10	D
1.7.11	A	1.7.12	D	1.7.13	D	1.7.14	D	1.7.15	B
1.8.1	A	1.8.2	B	1.8.3	D	1.8.4	A	1.8.5	A
1.8.6	16	1.8.7	A	1.9.1	N/A	1.9.2	13	1.9.3	C
1.9.4	D	1.10.1	2.33	1.10.2	D	1.10.3	225	1.10.4	B
1.11.1	N/A	1.11.2	N/A	1.11.3	C	1.11.4	A	1.11.5	N/A
1.11.6	C	1.11.7	C	1.11.8	N/A	1.11.9	C	1.11.10	D
1.11.11	A	1.11.12	B	1.11.13	D	1.11.14	C	1.11.15	C
1.11.16	D	1.11.17	D	1.11.18	D	1.11.19	B	1.11.20	C
1.11.21	B	1.11.22	D	1.11.23	B	1.11.24	A	1.11.25	9
1.11.26	A	1.11.27	D	1.11.28	51	1.11.29	0	1.11.30	C
1.11.31	26	1.11.32	D	1.11.33	81	1.11.34	1023 : 1023	1.11.35	15 : 15
1.11.36	D	1.11.37	C	1.11.38	C	1.11.39	D	1.11.40	C
1.12.1	147.1 : 148.1	1.13.1	4	1.13.2	D	1.13.3	3 : 3	1.13.4	C
1.14.1	C	1.14.2	N/A	1.14.3	N/A	1.14.4	C	1.14.5	0.08
1.15.1	N/A	1.15.2	N/A	1.15.3	N/A	1.15.4	N/A	1.15.5	N/A
1.15.6	N/A	1.15.7	N/A	1.15.8	B	1.15.9	A	1.15.10	N/A
1.15.11	B	1.15.12	N/A	1.15.13	B	1.15.14	C	1.15.15	B
1.15.16	D	1.15.17	A	1.15.18	B	1.15.19	D	1.15.20	X
1.15.21	A	1.15.22	D	1.15.23	A	1.15.24	A	1.15.25	B
1.15.26	2.32 : 2.33	1.15.27	B	1.15.28	A	1.15.29	C	1.15.30	C
1.15.31	B	1.15.32	C	1.15.33	A	1.16.1	C	1.16.2	10230
1.16.3	60 : 60	1.17.1	N/A	1.17.2	C	1.17.3	A	1.17.4	C
1.17.5	A	1.17.6	5	1.18.1	N/A	1.18.2	A	1.18.3	7
1.18.4	N/A	1.18.5	D	1.18.6	C	1.18.7	N/A	1.18.8	N/A

1.18.9	C	1.18.10	B	1.18.11	B	1.18.12	B	1.18.13	N/A
1.18.14	N/A	1.18.15	C	1.18.16	A	1.18.17	B	1.18.18	D
1.18.19	A	1.18.20	C	1.18.21	B	1.18.22	A	1.18.23	B
1.18.24	A	1.18.25	B	1.18.26	B	1.18.27	C	1.18.28	B
1.18.29	C	1.18.30	358	1.18.31	A	1.18.32	B	1.18.33	A
1.18.34	B	1.18.35	D	1.18.36	D	1.18.37	C	1.18.38	C
1.18.39	C	1.19.1	B	1.20.1	N/A	1.20.2	B;D;E	1.20.3	N/A
1.20.4	2	1.20.5	N/A	1.20.6	N/A	1.20.7	C	1.20.8	N/A
1.20.9	B	1.20.10	C	1.20.11	B	1.20.12	D	1.20.13	D
1.20.14	D	1.20.15	D	1.20.16	B	1.20.17	B	1.20.18	C
1.20.19	X	1.20.20	6	1.20.21	69	1.20.22	995	1.20.23	A
1.20.24	7	1.20.25	B	1.20.26	A	1.20.27	C	1.21.1	N/A
1.21.2	N/A	1.21.3	A	1.21.4	D	1.21.5	A	1.21.6	N/A
1.21.7	A	1.21.8	C	1.21.9	D	1.21.10	C	1.21.11	C
1.21.12	X	1.21.13	A	1.21.14	D	1.21.15	B	1.21.16	B
1.21.17	B	1.21.18	B	1.21.19	B	1.21.20	C	1.21.21	C
1.21.22	C	1.21.23	A	1.21.24	D	1.21.25	C	1.21.26	C
1.21.27	C	1.21.28	A	1.21.29	B				

2 Compiler Design (199)

Lexical analysis, Parsing, Syntax-directed translation, Runtime environments, Intermediate code generation.

Mark Distribution in Previous GATE

Year	2021-1	2021-2	2020	2019	2018	2017-1	2017-2	2016-1	2016-2	Minimum	Average	Maximum
1 Mark Count	1	2	2	2	1	2	2	1	1	1	1.5	2
2 Marks Count	3	2	1	2	2	2	1	3	2	1	2	3
Total Marks	7	6	4	6	5	6	4	7	5	**4**	**5.5**	**7**

2.1 Abstract Syntax Tree (1) top

2.1.1 Abstract Syntax Tree: GATE CSE 2015 Set 2 | Question: 14 top https://gateoverflow.in/8084

In the context of abstract-syntax-tree (AST) and control-flow-graph (CFG), which one of the following is TRUE?

A. In both AST and CFG, let node N_2 be the successor of node N_1. In the input program, the code corresponding to N_2 is present after the code corresponding to N_1
B. For any input program, neither AST nor CFG will contain a cycle
C. The maximum number of successors of a node in an AST and a CFG depends on the input program
D. Each node in AST and CFG corresponds to at most one statement in the input program

gate2015-cse-set2 compiler-design easy abstract-syntax-tree

Answer

Answers: Abstract Syntax Tree

2.1.1 Abstract Syntax Tree: GATE CSE 2015 Set 2 | Question: 14 top https://gateoverflow.in/8084

✔ Option (**C**) is Correct

A. is false , In CFG , code of N_2 may be present before N_1 when there is a loop or Goto.
B. is false , CFG contains cycle when input program has loop.
C. is true ,successors in AST and CFG depend on Input program.
D. is false, In CFG a single node may belong to a block of statements.

52 votes -- Himanshu Agarwal (12.4k points)

2.2 Assembler (9) top

2.2.1 Assembler: GATE CSE 1992 | Question: 01,viii top https://gateoverflow.in/553

The purpose of instruction location counter in an assembler is _______

gate1992 compiler-design assembler normal fill-in-the-blanks

Answer

2.2.2 Assembler: GATE CSE 1992 | Question: 03,ii top https://gateoverflow.in/579

Mention the pass number for each of the following activities that occur in a two pass assembler:

A. object code generation
B. literals added to literal table
C. listing printed
D. address resolution of local symbols

gate1992 compiler-design assembler easy match-the-following

Answer

2.2.3 Assembler: GATE CSE 1992 | Question: 3,i top https://gateoverflow.in/578

Write short answers to the following:

i. Which of the following macros can put a macro assembler into an infinite loop?

.MACRO MI,X .IF EQ,X M1 X+1 .ENDC .IF NE,X .WORD X .ENDC .ENDM	.MACRO M2,X .IF EQ,X M2 X .ENDC .IF NE,X .WORD X+1 .ENDC .ENDM

Give an example call that does so.

gate1992 compiler-design assembler normal descriptive

Answer

2.2.4 Assembler: GATE CSE 1993 | Question: 7.6 top

https://gateoverflow.in/2294

A simple two-pass assembler does the following in the first pass:

A. It allocates space for the literals.
B. It computes the total length of the program.
C. It builds the symbol table for the symbols and their values.
D. It generates code for all the load and store register instructions.
E. None of the above.

gate1993 compiler-design assembler easy multiple-selects

Answer

2.2.5 Assembler: GATE CSE 1994 | Question: 17a top

https://gateoverflow.in/2513

State whether the following statements are True or False with reasons for your answer:

Coroutine is just another name for a subroutine.

gate1994 compiler-design normal assembler true-false descriptive

Answer

2.2.6 Assembler: GATE CSE 1994 | Question: 17b top

https://gateoverflow.in/360171

State whether the following statements are True or False with reasons for your answer:

A two pass assembler uses its machine opcode table in the first pass of assembly.

gate1994 compiler-design normal assembler true-false descriptive

Answer

2.2.7 Assembler: GATE CSE 1994 | Question: 18a top

https://gateoverflow.in/2514

State whether the following statements are True or False with reasons for your answer

A subroutine cannot always be used to replace a macro in an assembly language program.

gate1994 compiler-design normal assembler true-false descriptive

Answer

2.2.8 Assembler: GATE CSE 1994 | Question: 18b top

https://gateoverflow.in/360170

State whether the following statements are True or False with reasons for your answer

A symbol declared as 'external' in an assembly language program is assigned an address outside the program by the assembler itself.

gate1994 compiler-design normal assembler true-false descriptive

Answer

2.2.9 Assembler: GATE CSE 1996 | Question: 1.17 top

https://gateoverflow.in/2721

The pass numbers for each of the following activities

i. object code generation
ii. literals added to literal table
iii. listing printed
iv. address resolution of local symbols that occur in a two pass assembler

respectively are

A. $1, 2, 1, 2$
B. $2, 1, 2, 1$
C. $2, 1, 1, 2$
D. $1, 2, 2, 2$

gate1996 compiler-design normal assembler

Answer

Answers: Assembler

2.2.1 Assembler: GATE CSE 1992 | Question: 01,viii top

https://gateoverflow.in/553

Each section of an assembler language program has a location counter used to assign storage addresses to your program's statements. As the instructions of a source module are being assembled, the location counter keeps track of the current location in storage.

13 votes -- Rohan Ghosh (1.6k points)

2.2.2 Assembler: GATE CSE 1992 | Question: 03,ii top

https://gateoverflow.in/579

✔
a. 2
b. 1
c. 2
d. 1

P.S. : In first pass, symbol table is created and In second pass, machine code is generated. Listing of final machine code is done after 2^{nd} pass only.

19 votes -- Aditya Gaurav (2.4k points)

2.2.3 Assembler: GATE CSE 1992 | Question: 3,i top

https://gateoverflow.in/578

✔ Macro M_2 can put the macro assembler into an infinite loop when called with $M_2(0)$

$(EQ(X)$ is TRUE when $X = 0)$

For M_1 the argument X is incremented for the recursive call and so the macro expansion will happen maximum 2 times.

7 votes -- Arjun Suresh (330k points)

2.2.4 Assembler: GATE CSE 1993 | Question: 7.6 top

https://gateoverflow.in/2294

✔ A, B, C are TRUE.

https://gateoverflow.in/?qa=blob&qa_blobid=2337905098612945492

References

👍 29 votes -- Arjun Suresh (330k points)

2.2.5 Assembler: GATE CSE 1994 | Question: 17a top

☛ https://gateoverflow.in/2513

True. The subroutine is a special case of a co-routine. A co-routine is a generalized form of a subroutine which is non-preemptive multitasking.

https://en.wikipedia.org/wiki/Coroutine

References

👍 8 votes -- srestha (85.2k points)

2.2.6 Assembler: GATE CSE 1994 | Question: 17b top

☛ https://gateoverflow.in/360171

✔ In the first pass of the assembler, Machine Opcode Table (MOT) is used to get the opcode size of the mnemonics which is needed to increment the LOCCTR which is needed to determine the address of the labels. The labels and their corresponding addresses go to the Symbol Table which is thus populated during the first phase and used in the second phase whereas MOT is not modified but just used in the two phases – in the second phase to convert menmonic to opcode.

👍 1 votes -- Arjun Suresh (330k points)

2.2.7 Assembler: GATE CSE 1994 | Question: 18a top

☛ https://gateoverflow.in/2514

✔ TRUE.

A macro is evaluated at compile time whereas a function call happens at runtime. So, we can write a macro to rename any symbol which is not possible to be replaced by a simple subroutine call. For example consider the following C code.

```
#define type int
type foo (type arg1)
{
    ...
}
#undef type
```

In the above code a macro is used to define a type which is used as the return and argument types for the function foo. This is not possible to be implemented as a simple subroutine call (but can be done using typedef is the language supports it).

👍 1 votes -- gatecse (62.6k points)

2.2.8 Assembler: GATE CSE 1994 | Question: 18b top

☛ https://gateoverflow.in/360170

✔ extern symbol in an assembler (or C) compilation unit (a file and all its included ones) is used to refer to global symbols (either variables or functions) in other parts of the program including any shared libraries.

Now, an assembler at the time of assembling has no information about the address of these extern symbols. It is the job of the linker to resolve them once assembling is over.

So, FALSE.

👍 1 votes -- gatecse (62.6k points)

2.2.9 Assembler: GATE CSE 1996 | Question: 1.17 top

☛ https://gateoverflow.in/2721

✔ Correct Option: **B**

The functions performed in pass 1 and pass 2 in 2 pass assembler are

Pass 1

1. Assign addresses to all statements in the program.
2. Save the values assigned to all labels for use in pass 2

3. Perform some processing of assembler directives.

Pass 2

1. Assemble instructions.
2. Generate data values defined by BYTE, WORD etc.
3. Perform processing of assembler directives not done during pass 1.
4. Write the program and the assembling listing

34 votes -- minal (13.1k points)

2.3 Code Optimization (7) top

2.3.1 Code Optimization: GATE CSE 2008 | Question: 12 top https://gateoverflow.in/410

Some code optimizations are carried out on the intermediate code because

A. They enhance the portability of the compiler to the target processor
B. Program analysis is more accurate on intermediate code than on machine code
C. The information from dataflow analysis cannot otherwise be used for optimization
D. The information from the front end cannot otherwise be used for optimization

gate2008-cse normal code-optimization compiler-design

Answer

2.3.2 Code Optimization: GATE CSE 2014 Set 1 | Question: 17 top https://gateoverflow.in/1784

Which one of the following is **FALSE**?

A. A basic block is a sequence of instructions where control enters the sequence at the beginning and exits at the end.
B. Available expression analysis can be used for common subexpression elimination.
C. Live variable analysis can be used for dead code elimination.
D. $x = 4 * 5 \Rightarrow x = 20$ is an example of common subexpression elimination.

gate2014-cse-set1 compiler-design code-optimization normal

Answer

2.3.3 Code Optimization: GATE CSE 2014 Set 3 | Question: 11 top https://gateoverflow.in/2045

The minimum number of arithmetic operations required to evaluate the polynomial $P(X) = X^5 + 4X^3 + 6X + 5$ for a given value of X, using only one temporary variable is ______.

gate2014-cse-set3 compiler-design numerical-answers normal code-optimization

Answer

2.3.4 Code Optimization: GATE CSE 2014 Set 3 | Question: 34 top https://gateoverflow.in/2068

Consider the basic block given below.

```
a = b + c
c = a + d
d = b + c
e = d - b
a = e + b
```

The minimum number of nodes and edges present in the DAG representation of the above basic block respectively are

A. 6 and 6
B. 8 and 10
C. 9 and 12
D. 4 and 4

gate2014-cse-set3 compiler-design code-optimization normal

Answer

2.3.5 Code Optimization: GATE CSE 2021 Set 1 | Question: 50 top

https://gateoverflow.in/357401

Consider the following C code segment:

```
a = b + c;
e = a + 1;
d = b + c;
f = d + 1;
g = e + f;
```

In a compiler, this code segment is represented internally as a directed acyclic graph (DAG). The number of nodes in the DAG is _____________

gate2021-cse-set1 compiler-design code-optimization dag

Answer

2.3.6 Code Optimization: GATE CSE 2021 Set 2 | Question: 30 top

https://gateoverflow.in/357510

Consider the following ANSI C code segment:

```
z=x + 3 + y->f1 + y->f2;
for (i = 0; i < 200; i = i + 2)
{
    if (z > i)
    {
        p = p + x + 3;
        q = q + y->f1;
    } else
    {
        p = p + y->f2;
        q = q + x + 3;
    }
}
```

Assume that the variable y points to a struct (allocated on the heap) containing two fields f1 and f2, and the local variables x, y, z, p, q, and i are allotted registers. Common sub-expression elimination (CSE) optimization is applied on the code. The number of addition and the dereference operations (of the form y ->f1 or y ->f2) in the optimized code, respectively, are:

A. 403 and 102
B. 203 and 2
C. 303 and 102
D. 303 and 2

gate2021-cse-set2 code-optimization compiler-design

Answer

2.3.7 Code Optimization: GATE CSE 2021 Set 2 | Question: 38 top

https://gateoverflow.in/357502

For a statement S in a program, in the context of liveness analysis, the following sets are defined:

$\text{USE}(S)$: the set of variables used in S

$\text{IN}(S)$: the set of variables that are live at the entry of S

$\text{OUT}(S)$: the set of variables that are live at the exit of S

Consider a basic block that consists of two statements, S_1 followed by S_2. Which one of the following statements is correct?

A. $\text{OUT}(S_1) = \text{IN }(S_2)$
B. $\text{OUT }(S_1) = \text{IN }(S_1) \cup \text{ USE }(S_1)$
C. $\text{OUT }(S_1) = \text{IN }(S_2) \: \cup \: \text{OUT }(S_2)$
D. $\text{OUT }(S_1) = \text{USE }(S_1) \cup \text{IN }(S_2)$

gate2021-cse-set2 code-optimization compiler-design

Answer

Answers: Code Optimization

2.3.1 Code Optimization: GATE CSE 2008 | Question: 12 top https://gateoverflow.in/410

✔ Answer: A

Option (B) is also true. But the main purpose of doing some code-optimization on intermediate code generation is to enhance the portability of the compiler to target processors. So Option A) is more suitable here. Intermediate code is machine/architecture independent code. So a compiler can optimize it without worrying about the architecture on which the code is going to execute (it may be the same or the other). So that kind of compiler can be used by multiple different architectures. In contrast to that, suppose code optimization is done on target code, which is machine/architecture dependent, then the compiler has be specific about the optimizations on that kind of code. In this case the compiler can't be used by multiple different architectures, because the target code produced on different architectures would be different. Hence portability reduces here.

ref-http://quiz.geeksforgeeks.org/code-generation-and-optimization/

References

30 votes -- vnc (2.1k points)

2.3.2 Code Optimization: GATE CSE 2014 Set 1 | Question: 17 top https://gateoverflow.in/1784

✔

A. A basic block is a sequence of instructions where control enters the sequence at the beginning and exits at the end is TRUE.

B. Available expression analysis can be used for common subexpression elimination is TRUE. Available expressions is an analysis algorithm that determines for each point in the program the set of expressions that need not be recomputed. Available expression analysis is used to do global common subexpression elimination (CSE). If an expression is available at a point, there is no need to re-evaluate it.

C. Live variable analysis can be used for dead code elimination is TRUE.

D. $x = 4 * 5 \Rightarrow x = 20$ is an example of common subexpression elimination is FALSE. Common subexpression elimination (CSE) refers to compiler optimization replaces identical expressions (i.e., they all evaluate to the same value) with a single variable holding the computed value when it is worthwhile to do so Source: Geeksforgeeks

Correct Answer: D

28 votes -- Pyuri sahu (1.5k points)

2.3.3 Code Optimization: GATE CSE 2014 Set 3 | Question: 11 top https://gateoverflow.in/2045

✔ $P(X) = x^5 + 4x^3 + 6x + 5$

$= x(x^4 + 4x^2 + 6) + 5$

$= x(x(x^3 + 4x) + 6) + 5$

$= x(x(x(x^2 + 4)) + 6) + 5$

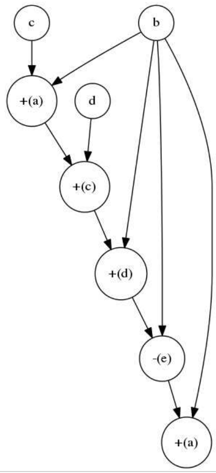
$= x(x(x(x(x) + 4)) + 6) + 5$

mul = pair of brackets 4
add = num of signs 3
total 7

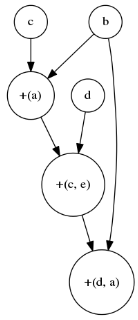
90 votes -- jayendra (6.7k points)

2.3.4 Code Optimization: GATE CSE 2014 Set 3 | Question: 34 top https://gateoverflow.in/2068

✔ A normal DAG construction will give 8 nodes and 10 edges as shown below.



Since, this question asks for minimum possible, we can assume algebraic simplification is allowed. So, $d = b + c, e = d - b$; can be simplified to $d = b + c$; $e = c$; Similarly, $e = d - b$; $a = e + b$; can be simplified to $a = d$. This gives the following DAG with 6 nodes and 6 edges.



Reference: https://cs.nyu.edu/~gottlieb/courses/2000s/2006-07-fall/compilers/lectures/lecture-14.html

Correct Answer: A

References

65 votes -- Arjun Suresh (330k points)

2.3.5 Code Optimization: GATE CSE 2021 Set 1 | Question: 50 top https://gateoverflow.in/357401

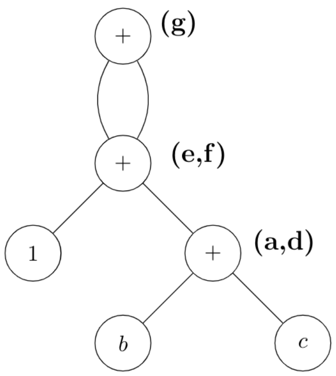
✔ Here a and d are same as both add same values (bc) (common sub-expression elimination)

Since a and d are same f and e are also same as they compute $a + 1$ and $d + 1$ respectively.

- $a = d = b + c$
- $e = f = a + 1$
- $g = e + e$ (f and e being same)

So total no of nodes is 6 $(a, b, c, e, 1, g)$

Ans : 6 nodes



👍 3 votes -- Ankur tiwari (557 points)

2.3.6 Code Optimization: GATE CSE 2021 Set 2 | Question: 30 top

☛ https://gateoverflow.in/357510

✔

```
t1 = x + 3 // 1 addition
t2 = y->f1; // 1 dereference
t3 = y->f2; // 1 dereference
z = t1 + t2 + t3 // 2 additions
for (i = 0; i < 200; i += 2) {
    if (z > i) {
        p = p + t1; // 1 addition
        q = q + t2; // 1 addition
    } else {
        p = p + t3; // 1 addition
        q = q + t1; // 1 addition
    }
}
```

Whether we take if or else block we get 2 additions, the loop runs exactly $\frac{200}{2} = 100$ times, so from loop we get $2 \times 100 = 200$ additions plus 100 additions for incrementing the value of i, before loop we had perform 3 additions, so total additions 303.

We only do two de-reference outside the for loop, so total de-references $= 2$.

Option D.

👍 5 votes -- zxy123 (2.5k points)

```
t1 = x + 3;        // 1 addition
t2 = y -> f1;       // 1 dereference
t3 = y -> f2;       // 1 dereference
z = t1 + t2 + t3; // 2 additions
for (i = 0; i < 200; i += 2) { // 100 additions
    if (z > i) {
        p = p + t1; // 1 addition
        q = q + t2; // 1 addition
    } else {
        p = p + t3; // 1 addition
        q = q + t1; // 1 addition
    }
}
```

So, in total we get $1 + 2 + 100 + 100 * 2 = 303$ additions and 2 defreferences. Since all the variables are mentioned to be in registers and any way p and q are not struct objects there is no pointer aliasing issue (say if y was pointing to object p or q, we cannot move the sub expression out of the loop – they are no longer loop invariant.

Option D

Not asked in this question. But lets do some more optimizations here.

```
t1 = x + 3;
t2 = y -> f1;
t3 = y -> f2;
z = t1 + t2 + t3;

for (i = z+1 + (z%2); i < 200; i += 2) {
        p = p + t1;
        q = q + t2;
}

```

```
    for (i = 0; i <= z; i += 2) {
            p = p + t3;
            q = q + t1;
        }
    }
```

The above optimization is loop splitting. The advantage here is now we have one less branch inside the loop – less chance of branch miss prediction and more expected instruction level parallelism – remember pipeline stalls due to branch instructions in COA. Also, now we can optimize the code even further as follows:

```
t1 = x + 3;
t2 = y -> f1;
t3 = y -> f2;
z = t1 + t2 + t3;

p = p + ((200-z-1-(z%2))/2) * t1;
q = q + ((200-z-1-(z%2))/2) * t2;

p = p + ((z+1)/2) * t3;
q = q + ((z+1)/2)* t1;
```

Again doing sub-expression elimination:

```
t1 = x + 3;
t2 = y -> f1;
t3 = y -> f2;
z = t1 + t2 + t3;

t4 = ((200-z-1-(z%2))/2);
p = p + t4 * t1;
q = q + t4 * t2;
//previous t4 usage is dead here
t4 = ((z+1)/2);
p = p + t4 * t3;
q = q + t4 * t1;
```

So, finally, 4 multiplications, 3 divisions/mod, 11 additions/subtractions.

That's what compiler does freely for you :)

4 votes -- Arjun Suresh (330k points)

2.3.7 Code Optimization: GATE CSE 2021 Set 2 | Question: 38 top

https://gateoverflow.in/357502

✔ When a basic block S_2 immediately follows another basic block S_1, all variables that are live at exit of S_1 must be live at entry of S_2 (no intermediate place where they can get killed) and no other variable can be live at entry of S_2 as a basic block is always **single entry** and **single exit**.

So, $\text{OUT}(S_1) = \text{IN}\ (S_2)$

Correct option: A

Liveness Analysis Slides

References

0 votes -- Arjun Suresh (330k points)

2.4 Compilation Phases (10) top

2.4.1 Compilation Phases: GATE CSE 1987 | Question: 1-xi top

https://gateoverflow.in/80282

In a compiler the module the checks every character of the source text is called:

A. The code generator.
B. The code optimiser.
C. The lexical analyser.
D. The syntax analyser.

gate1987 compiler-design compilation-phases

Answer

2.4.2 Compilation Phases: GATE CSE 1990 | Question: 2-ix top https://gateoverflow.in/84033

Match the pairs in the following questions:

(a)	Lexical analysis	(p)	DAG's
(b)	Code optimization	(q)	Syntax trees
(c)	Code generation	(r)	Push down automaton
(d)	Abelian groups	(s)	Finite automaton

gate1990 match-the-following compiler-design compilation-phases

Answer

2.4.3 Compilation Phases: GATE CSE 2005 | Question: 61 top https://gateoverflow.in/4066

Consider line number 3 of the following C-program.

```
int main() {                /*Line 1 */
    int I, N;               /*Line 2 */
    fro (I=0, I<N, I++); /*Line 3 */
}
```

Identify the compiler's response about this line while creating the object-module:

A. No compilation error
B. Only a lexical error
C. Only syntactic errors
D. Both lexical and syntactic errors

gate2005-cse compiler-design compilation-phases normal

Answer

2.4.4 Compilation Phases: GATE CSE 2009 | Question: 17 top https://gateoverflow.in/1309

Match all items in Group 1 with the correct options from those given in Group 2.Syntax analysis

Group 1		Group 2	
P.	Regular Expression	1.	Syntax analysis
Q.	Pushdown automata	2.	Code generation
R.	Dataflow analysis	3.	Lexical analysis
S.	Register allocation	4.	Code optimization

A. P-4, Q-1, R-2, S-3
B. P-3, Q-1, R-4, S-2
C. P-3, Q-4, R-1, S-2
D. P-2, Q-1, R-4, S-3

gate2009-cse compiler-design easy compilation-phases

Answer

2.4.5 Compilation Phases: GATE CSE 2015 Set 2 | Question: 19 top https://gateoverflow.in/8098

Match the following:

P.	Lexical analysis	1.	Graph coloring
Q.	Parsing	2.	DFA minimization
R.	Register allocation	3.	Post-order traversal
S.	Expression evaluation	4.	Production tree

A. P-2, Q-3, R-1, S-4

B. P-2, Q-1, R-4, S-3
C. P-2, Q-4, R-1, S-3
D. P-2, Q-3, R-4, S-1

gate2015-cse-set2 compiler-design normal compilation-phases

Answer

2.4.6 Compilation Phases: GATE CSE 2016 Set 2 | Question: 19 top

https://gateoverflow.in/39548

Match the following:

(P)	Lexical analysis	(i)	Leftmost derivation
(Q)	Top down parsing	(ii)	Type checking
(R)	Semantic analysis	(iii)	Regular expressions
(S)	Runtime environment	(iv)	Activation records

A. P $\leftrightarrow$ i, Q $\leftrightarrow$ ii, R $\leftrightarrow$ iv, S $\leftrightarrow$ iii
B. P $\leftrightarrow$ iii, Q $\leftrightarrow$ i, R $\leftrightarrow$ ii, S $\leftrightarrow$ iv
C. P $\leftrightarrow$ ii, Q $\leftrightarrow$ iii, R $\leftrightarrow$ i, S $\leftrightarrow$ iv
D. P $\leftrightarrow$ iv, Q $\leftrightarrow$ i, R $\leftrightarrow$ ii, S $\leftrightarrow$ iii

gate2016-cse-set2 compiler-design easy compilation-phases

Answer

2.4.7 Compilation Phases: GATE CSE 2017 Set 2 | Question: 05 top

https://gateoverflow.in/118592

Match the following according to input (from the left column) to the compiler phase (in the right column) that processes it:

P. Syntax tree	i. Code generator
Q. Character stream	ii. Syntax analyser
R. Intermediate representation	iii. Semantic analyser
S. Token stream	iv. Lexical analyser

A. P-ii; Q-iii; R-iv; S-i
B. P-ii; Q-i; R-iii; S-iv
C. P-iii; Q-iv; R-i; S-ii
D. P-i; Q-iv; R-ii; S-iii

gate2017-cse-set2 compiler-design compilation-phases easy

Answer

2.4.8 Compilation Phases: GATE CSE 2018 | Question: 8 top

https://gateoverflow.in/204082

Which one of the following statements is FALSE?

A. Context-free grammar can be used to specify both lexical and syntax rules
B. Type checking is done before parsing
C. High-level language programs can be translated to different Intermediate Representations
D. Arguments to a function can be passed using the program stack

gate2018-cse compiler-design easy compilation-phases

Answer

2.4.9 Compilation Phases: GATE CSE 2020 | Question: 9 top

https://gateoverflow.in/333222

Consider the following statements.

I. Symbol table is accessed only during lexical analysis and syntax analysis.
II. Compilers for programming languages that support recursion necessarily need heap storage for memory allocation in the

run-time environment.

III. Errors violating the condition '*any variable must be declared before its use*' are detected during syntax analysis.

Which of the above statements is/are TRUE?

A. I only
B. I and III only
C. II only
D. None of I, II and III

gate2020-cse compiler-design compilation-phases runtime-environments

Answer

2.4.10 Compilation Phases: GATE CSE 2021 Set 2 | Question: 3 top https://gateoverflow.in/357537

Consider the following ANSI C program:

```
int main () {
    Integer x;
    return 0;
}
```

Which one of the following phases in a seven-phase C compiler will throw an error?

A. Lexical analyzer
B. Syntax analyzer
C. Semantic analyzer
D. Machine dependent optimizer

gate2021-cse-set2 compilation-phases compiler-design

Answer

Answers: Compilation Phases

2.4.1 Compilation Phases: GATE CSE 1987 | Question: 1-xi top https://gateoverflow.in/80282

✔ lexical analyser phase checks every character of text to identify tokens..

21 votes -- kirti singh (2.6k points)

2.4.2 Compilation Phases: GATE CSE 1990 | Question: 2-ix top https://gateoverflow.in/84033

✔

(a)	Lexical analysis	(s)	Finite automaton (DFA creation for finding tokens)
(b)	Code optimization	(p)	DAG's (Common subtree minimization)
(c)	Code generation	(q)	Syntax trees (one can construct a derivation and from it a parse tree that can be used for code generation)
(d)	Abelian groups	(r)	Push down automaton

45 votes -- Prashant Singh (47.1k points)

2.4.3 Compilation Phases: GATE CSE 2005 | Question: 61 top https://gateoverflow.in/4066

✔ C language allows only certain words in it- these are called tokens. If we input any invalid tokens it causes lexical error.

eg:
$44a44$
causes lexical error as in C as an alphabet cannot come in between digits.

Syntactic error is caused by bad combination of tokens. For example, we cannot have a constant on the left hand side of an assignment statement, a for loop must have two expressions inside $()$ separated by semi colon etc.

In the given question, line 3 won't cause a lexical error or syntactic error. The statement will be treated as a function call with three arguments. Function definition being absent will cause link time error, but the question asks only for compile-time errors. So, (a) must be the answer.

PS: Implicit function declaration was removed from $C99$ standard onwards. As per current standard, we should not use a function without declaration. Still, we cannot guarantee "compilation error"- just expect compiler warnings in C. In C++ this should produce a compilation (semantic) error. The output of compiling the above code using different standards are given below:

```
arjun@linux:~$ gcc -c chk.c
chk.c: In function 'main':
chk.c:3:2: warning: implicit declaration of function 'fro' [-Wimplicit-function-declaration]
  fro (I=0, I<N, I++); /*Line 3 */
  ^
arjun@linux:~$ gcc -c -ansi chk.c
arjun@linux:~$ gcc -c -std=c99 chk.c
chk.c: In function 'main':
chk.c:3:2: warning: implicit declaration of function 'fro' [-Wimplicit-function-declaration]
  fro (I=0, I<N, I++); /*Line 3 */
  ^
arjun@linux:~$ gcc -c -std=c11 chk.c
chk.c: In function 'main':
chk.c:3:2: warning: implicit declaration of function 'fro' [-Wimplicit-function-declaration]
  fro (I=0, I<N, I++); /*Line 3 */
```

http://stackoverflow.com/questions/15570553/lexical-and-semantic-errors-in-c

References

92 votes -- Arjun Suresh (330k points)

2.4.4 Compilation Phases: GATE CSE 2009 | Question: 17 top https://gateoverflow.in/1309

✔ Correct Option: **B**

Regular expressions are used in lexical analysis.

Pushdown automata is related to context free grammar which is related to syntax analysis.

Dataflow analysis is done in code optimization.

Register allocation is done in code generation.

44 votes -- Keith Kr (4.5k points)

2.4.5 Compilation Phases: GATE CSE 2015 Set 2 | Question: 19 top https://gateoverflow.in/8098

✔

1. Regular expression uses FA & Regular Sets.
2. Expression can be evaluated with postfix Traversals.
3. Register allocation can be modeled by graph coloring.
4. The parser constructs a production tree.

So, answer is C.

33 votes -- shekhar chauhan (32.8k points)

2.4.6 Compilation Phases: GATE CSE 2016 Set 2 | Question: 19 top https://gateoverflow.in/39548

✔ Correct Option: **B**

- Lexical Analysis phase uses regular expressions.
- LMD is involved in top down parsing.
- Type checking is done in semantic analysis phase.
- Activation records are related to Run Time Environments.

36 votes -- Sharathkumar Anbu (595 points)

2.4.7 Compilation Phases: GATE CSE 2017 Set 2 | Question: 05 top https://gateoverflow.in/118592

✔ Correct Option: **C**

- Q - iv because Character stream is given as input to lexical analyser
- P - iii Syntax tree is given as input to semantic analyser
- R - i Intermediate code given as input to code generator
- S - ii Token stream given as input to syntax analyser

32 votes -- Arnabi Bej (5.8k points)

2.4.8 Compilation Phases: GATE CSE 2018 | Question: 8 top https://gateoverflow.in/204082

✔ A. Since Lexical rules are nothing but regular expressions, we can use CFGs to represent such rules.(Every Type-3 grammar is Type-2 grammar) Additionally, syntax rules can be represented by CFGs. (True)

B. Type checking is done during Semantic Analysis phase which comes after Parsing. (False)

C. We have various types of Intermediate Code Representations, ex 3-address code, Postfix notation, Syntax trees. (True)

D. Program stack holds the activation record of the function called, which stores function parameters, return value, return address etc.(True)

Correct Answer: B

37 votes -- Neelay Upadhyaya (1.1k points)

2.4.9 Compilation Phases: GATE CSE 2020 | Question: 9 top https://gateoverflow.in/333222

✔ 1. False.

> The **symbol table is accessed** by most phases of a compiler, beginning with **lexical analysis**, and continuing through optimization.

Symbol table is accessed during other stages also.

Ref: https://en.m.wikipedia.org/wiki/Symbol_table

2. Not essential, any one of heap and stack is enough to support recursion. Dynamic allocation of activation records is essential to implement recursion. Remember the stack size can also grow dynamical (see C memory layout).

3. Syntax analyser uses CFL which cannot check for this, we need power of Context sensitive language which is available in semantic analysis phase. So this error is detected only during semantic analysis phase.

So D is correct.

References

4 votes -- Subham Mishra (11.4k points)

2.4.10 Compilation Phases: GATE CSE 2021 Set 2 | Question: 3 top https://gateoverflow.in/357537

✔ This question is difficult to answer from a practical point of view because most of the C compilers (even other language compilers) do not follow the classical ordering of compilation phases. Since this is a one-mark question ignoring the practical implementations and going by just theory answer will be syntax error. Because there are no lexical errors and "Integer" and "x" get read as identifiers as shown in the following output.

```
arjun@Armi:~$ cat p1.c
int main()
{
 Integer x;
 return 0;
}
arjun@Armi:~$ clang p1.c -c -Xclang -dump-tokens
int 'int'  [StartOfLine] Loc=<p1.c:1:1>
identifier 'main'  [LeadingSpace] Loc=<p1.c:1:5>
```

```
l_paren '('  Loc=<p1.c:1:9>
r_paren ')'  Loc=<p1.c:1:10>
l_brace '{'  [StartOfLine] Loc=<p1.c:2:1>
identifier 'Integer'  [StartOfLine] [LeadingSpace] Loc=<p1.c:3:2>
identifier 'x'  [LeadingSpace] Loc=<p1.c:3:10>
semi ';'  Loc=<p1.c:3:11>
return 'return'  [StartOfLine] [LeadingSpace] Loc=<p1.c:4:2>
numeric_constant '0'  [LeadingSpace] Loc=<p1.c:4:9>
semi ';'  Loc=<p1.c:4:10>
r_brace '}'  [StartOfLine] Loc=<p1.c:5:1>
eof ''  Loc=<p1.c:5:2>
```

Now, when this stream of tokens get passed to the syntax analyser – we have an identifier followed by another identifier which is not valid in C syntax – so **syntax error**. And this must be the answer here though we can argue for semantic error as well as follows.

Now consider a typedef usage like "typedef int Integer". Now, this can be implemented by the compiler in multiple ways. One option is to immediately change the token type of "Integer" from identifier to the given "type". Otherwise the syntax check can go with the AST generation. But if we go by the classical meaning of the compilation phases here we are matching a string which means it is a semantic phase.

Correct Answer: Syntax Analysis/Semantic analysis

More read: https://stackoverflow.com/questions/66290247/integer-x-is-syntactic-error-or-semantic-error

Official answer given in GATE key is "Semantic analysis" – but even the best compiler professors won't conclude on that.

Though this was a bad question and even worse answer key, lets use it to learn something useful.

The following three flags will force `cc` (C compiler) to check that your code complies to the relevant international standard, often referred to as the ANSI standard, though strictly speaking it is an ISO standard.

- `-Wall`
  Enable all the warnings which the authors of `cc` believe are worthwhile. Despite the name, it will not enable all the warnings `cc` is capable of.
- `-ansi`
  Turn off most, but not all, of the non-ANSI C features provided by `cc`. Despite the name, it does not guarantee strictly that your code will comply to the standard.
- `-pedantic`
  Turn off *all* `cc`'s non-ANSI C features.

Without these flags, `cc` will allow you to use some of its non-standard extensions to the standard. Some of these are very useful, but will not work with other compilers—in fact, one of the main aims of the standard is to allow people to write code that will work with any compiler on any system. This is known as *portable code*.

https://docs.freebsd.org/en_US.ISO8859-1/books/developers-handbook/tools-compiling.html

References

19 votes -- gatecse (62.6k points)

2.5 Expression Evaluation (2) top

2.5.1 Expression Evaluation: GATE CSE 2002 | Question: 2.19 top https://gateoverflow.in/849

To evaluate an expression without any embedded function calls

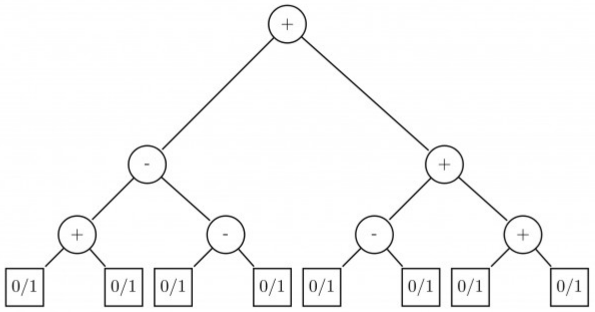
A. One stack is enough
B. Two stacks are needed
C. As many stacks as the height of the expression tree are needed
D. A Turing machine is needed in the general case

gate2002-cse compiler-design expression-evaluation easy

Answer

2.5.2 Expression Evaluation: GATE CSE 2014 Set 2 | Question: 39 top https://gateoverflow.in/1999

Consider the expression tree shown. Each leaf represents a numerical value, which can either be 0 or 1. Over all possible choices of the values at the leaves, the maximum possible value of the expression represented by the tree is ___.



gate2014-cse-set2 compiler-design normal expression-evaluation numerical-answers

Answer

Answers: Expression Evaluation

2.5.1 Expression Evaluation: GATE CSE 2002 | Question: 2.19 top https://gateoverflow.in/849

✔ Expression without any calls in it $\implies 1+2*3-4$

Expression with embedded calls $\implies 1+\text{fun1}(a,b,c)*\text{fun2}(3.4,58)-\text{fun3}(x,yz);$

First we can convert Infix to Postfix using single stack (Using it as operator stack)

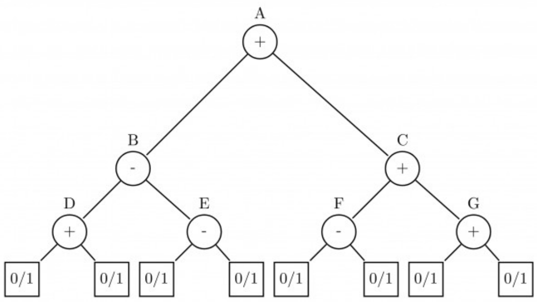
Then we can evaluate that expression using Single stack.

Correct Answer: A.

30 votes -- Akash Kanase (36k points)

2.5.2 Expression Evaluation: GATE CSE 2014 Set 2 | Question: 39 top https://gateoverflow.in/1999

✔



$A=B+C$

For A to be maximum, both B and C should be maximum

$B=D-E$

For B to be maximum, D should be maximum and E should be minimum

$C=F+G$

For C to be maximum, both F and G should be maximum

The maximum value of D is 2 $(1+1=2)$

The minimum value of E is -1 $(0-1=-1)$

The maximum value of F is 1 $(1-0=1)$

The maximum value of G is 2 $(1+1=2)$

$B=2-(-1)=2+1=3$

$C = 1 + 2 = 3$

$A = B + C = 3 + 3 = 6$

6 is the answer

16 votes -- Pankaj Kumar (7.8k points)

2.6 Grammar (42) top

2.6.1 Grammar: GATE CSE 1990 | Question: 16a top https://gateoverflow.in/86869

Show that grammar G_1 is ambiguous using parse trees:

$G_1 : S \rightarrow$ if S then S else S

$S \rightarrow$ if S then S

gate1990 descriptive compiler-design grammar

Answer

2.6.2 Grammar: GATE CSE 1991 | Question: 10a top https://gateoverflow.in/537

Consider the following grammar for arithmetic expressions using binary operators $-$ and $/$ which are not associative

- $E \rightarrow E - T \mid T$
- $T \rightarrow T/F \mid F$
- $F \rightarrow (E) \mid id$

(E is the start symbol)

Is the grammar unambiguous? Is so, what is the relative precedence between $-$ and $/$? If not, give an unambiguous grammar that gives $/$ precedence over $-$.

gate1991 grammar compiler-design normal descriptive

Answer

2.6.3 Grammar: GATE CSE 1991 | Question: 10b top https://gateoverflow.in/43604

Consider the following grammar for arithmetic expressions using binary operators $-$ and $/$ which are not associative

- $E \rightarrow E - T \mid T$
- $T \rightarrow T/F \mid F$
- $F \rightarrow (E) \mid id$

(E is the start symbol)

Does the grammar allow expressions with redundant parentheses as in (id/id) or in $id - (id/id)$? If so, convert the grammar into one which does not generate expressions with redundant parentheses. Do this with minimum number of changes to the given production rules and adding at most one more production rule.

gate1991 grammar compiler-design normal descriptive

Answer

2.6.4 Grammar: GATE CSE 1991 | Question: 10c top https://gateoverflow.in/43605

Consider the following grammar for arithmetic expressions using binary operators $-$ and $/$ which are not associative

- $E \rightarrow E - T \mid T$
- $T \rightarrow T/F \mid F$
- $F \rightarrow (E) \mid id$

(E is the start symbol)

Does the grammar allow expressions with redundant parentheses as in (id/id) or in $id - (id/id)$? If so, convert the grammar into one which does not generate expressions with redundant parentheses. Do this with minimum number of changes to the given production rules and adding at most one more production rule.

Convert the grammar obtained above into one that is not left recursive.

gate1991 grammar compiler-design normal descriptive

Answer

2.6.5 Grammar: GATE CSE 1994 | Question: 1.18 top

https://gateoverflow.in/2461

Which of the following features cannot be captured by context-free grammars?

A. Syntax of if-then-else statements
B. Syntax of recursive procedures
C. Whether a variable has been declared before its use
D. Variable names of arbitrary length

gate1994 compiler-design grammar normal

Answer

2.6.6 Grammar: GATE CSE 1994 | Question: 20 top

https://gateoverflow.in/2516

A grammar G is in Chomsky-Normal Form (CNF) if all its productions are of the form $A \to BC$ or $A \to a$, where A, B and C, are non-terminals and a is a terminal. Suppose G is a CFG in CNF and w is a string in $L(G)$ of length n, then how long is a derivation of w in G?

gate1994 compiler-design grammar normal descriptive

Answer

2.6.7 Grammar: GATE CSE 1994 | Question: 3.5 top

https://gateoverflow.in/2482

Match the following items

(i)	Backus-Naur form	(a)	Regular expressions
(ii)	Lexical analysis	(b)	LALR(1) grammar
(iii)	YACC	(c)	LL(1) grammars
(iv)	Recursive descent parsing	(d)	General context-free grammars

gate1994 compiler-design grammar normal match-the-following

Answer

2.6.8 Grammar: GATE CSE 1995 | Question: 1.10 top

https://gateoverflow.in/2597

Consider a grammar with the following productions

- $S \to a\alpha b \mid b\alpha c \mid aB$
- $S \to \alpha S \mid b$
- $S \to \alpha bb \mid ab$
- $S\alpha \to bdb \mid b$

The above grammar is:

A. Context free
B. Regular
C. Context sensitive
D. $LR(k)$

gate1995 compiler-design grammar normal

Answer

2.6.9 Grammar: GATE CSE 1995 | Question: 9 top

https://gateoverflow.in/2644

A. Translate the arithmetic expression $a^* - (b + c)$ into syntax tree.

B. A grammar is said to have cycles if it is the case that $A \overset{+}{\Rightarrow} A$
Show that no grammar that has cycles can be $\text{LL}(1)$.

gate1995 compiler-design grammar normal descriptive

Answer

2.6.10 Grammar: GATE CSE 1996 | Question: 11 top

https://gateoverflow.in/2763

Let G be a context-free grammar where $G = (\{S, A, B, C\}, \{a, b, d\}, P, S)$ with the productions in P given below.

- $S \rightarrow ABAC$
- $A \rightarrow aA \mid \varepsilon$
- $B \rightarrow bB \mid \varepsilon$
- $C \rightarrow d$

(ε denotes the null string). Transform the grammar G to an equivalent context-free grammar G' that has no ε productions and no unit productions. (A unit production is of the form $x \rightarrow y$, and x and y are non terminals).

gate1996 compiler-design grammar normal descriptive

Answer

2.6.11 Grammar: GATE CSE 1996 | Question: 2.10 top

https://gateoverflow.in/2739

The grammar whose productions are

- $\langle \text{stmt} \rangle \rightarrow$ if id then $\langle \text{stmt} \rangle$
- $\langle \text{stmt} \rangle \rightarrow$ if id then $\langle \text{stmt} \rangle$ else $\langle \text{stmt} \rangle$
- $\langle \text{stmt} \rangle \rightarrow$ id $:=$ id

is ambiguous because

(a) the sentence

```
if a then if b then c:= d
```

has more than two parse trees

(b) the left most and right most derivations of the sentence

```
if a then if b then c:= d
```

give rise to different parse trees

(c) the sentence

```
if a then if b then c:= d else c:= f
```

has more than two parse trees

(d) the sentence

```
if a then if b then c:= d else c:= f
```

has two parse trees

gate1996 compiler-design grammar normal

Answer

2.6.12 Grammar: GATE CSE 1997 | Question: 1.6 top

https://gateoverflow.in/2222

In the following grammar

- $X ::= X \oplus Y \mid Y$

- $Y ::= Z * Y \mid Z$
- $Z ::= id$

Which of the following is true?

A. ‘ $\oplus$ ’ is left associative while ‘ $*$ ’ is right associative
B. Both ‘ $\oplus$ ’ and ‘ $*$ ’ are left associative
C. ‘ $\oplus$ ’ is right associative while ‘ $*$ ’ is left associative
D. None of the above

gate1997 compiler-design grammar normal

Answer

2.6.13 Grammar: GATE CSE 1997 | Question: 11 top

https://gateoverflow.in/2271

Consider the grammar

- $S \rightarrow bSe$
- $S \rightarrow PQR$
- $P \rightarrow bPc$
- $P \rightarrow \varepsilon$
- $Q \rightarrow cQd$
- $Q \rightarrow \varepsilon$
- $R \rightarrow dRe$
- $R \rightarrow \varepsilon$

where S, P, Q, R are non-terminal symbols with S being the start symbol; b, c, d, e are terminal symbols and ‘ ε’ is the empty string. This grammar generates strings of the form b^i, c^j, d^k, e^m for some $i, j, k, m \geq 0$.

a. What is the condition on the values of i, j, k, m?
b. Find the smallest string that has two parse trees.

gate1997 compiler-design grammar normal theory-of-computation descriptive

Answer

2.6.14 Grammar: GATE CSE 1998 | Question: 14 top

https://gateoverflow.in/1728

A. Let $G_1 = (N, T, P, S_1)$ be a CFG where, $N = \{S_1, A, B\}, T = \{a, b\}$ and P is given by

$$\begin{array}{c|c} S_1 \rightarrow aS_1b & S_1 \rightarrow aBb \\ S_1 \rightarrow aAb & B \rightarrow Bb \\ A \rightarrow aA & B \rightarrow b \\ A \rightarrow a & \end{array}$$

What is $L(G_1)$?

B. Use the grammar in Part(a) to give a CFG for $L_2 = \{a^i b^j a^k b^l \mid i, j, k, l \geq 1, i = j \text{ or } k = l\}$ by adding not more than 5 production rules.
C. Is L_2 inherently ambiguous?

gate1998 compiler-design grammar descriptive

Answer

2.6.15 Grammar: GATE CSE 1998 | Question: 6b top

https://gateoverflow.in/1697

Consider the grammar

- S $\rightarrow Aa \mid b$
- A $\rightarrow Ac \mid Sd \mid \epsilon$

Construct an equivalent grammar with no left recursion and with minimum number of production rules.

gate1998 compiler-design grammar descriptive

Answer

2.6.16 Grammar: GATE CSE 1999 | Question: 2.15 top

https://gateoverflow.in/1493

A grammar that is both left and right recursive for a non-terminal, is

A. Ambiguous
B. Unambiguous
C. Information is not sufficient to decide whether it is ambiguous or unambiguous
D. None of the above

gate1999 compiler-design grammar normal

Answer

2.6.17 Grammar: GATE CSE 2000 | Question: 2.21, ISRO2015-24 top

https://gateoverflow.in/668

Given the following expression grammar:

$$E \rightarrow E * F \mid F + E \mid F$$

$$F \quad \rightarrow F - F \mid id$$

Which of the following is true?

A. $*$ has higher precedence than $+$
B. $-$ has higher precedence than $*$
C. $+$ and $-$ have same precedence
D. $+$ has higher precedence than $*$

gate2000-cse grammar normal compiler-design isro2015

Answer

2.6.18 Grammar: GATE CSE 2001 | Question: 1.18 top

https://gateoverflow.in/711

Which of the following statements is false?

A. An unambiguous grammar has same leftmost and rightmost derivation
B. An LL(1) parser is a top-down parser
C. LALR is more powerful than SLR
D. An ambiguous grammar can never be LR(k) for any k

gate2001-cse compiler-design grammar normal

Answer

2.6.19 Grammar: GATE CSE 2001 | Question: 18 top

https://gateoverflow.in/759

A. Remove left-recursion from the following grammar: $S \rightarrow Sa \mid Sb \mid a \mid b$
B. Consider the following grammar:

$S \rightarrow aSbS \mid bSaS \mid \epsilon$

Construct all possible parse trees for the string abab. Is the grammar ambiguous?

gate2001-cse compiler-design grammar descriptive

Answer

2.6.20 Grammar: GATE CSE 2003 | Question: 56 top

☛ https://gateoverflow.in/944

Consider the grammar shown below

- $S \to iEtSS' \mid a$
- $S' \to eS \mid \epsilon$
- $E \to b$

In the predictive parse table, M, of this grammar, the entries $M[S', e]$ and $M[S', \$]$ respectively are

A. $\{S' \to eS\}$ and $\{S' \to \epsilon\}$
B. $\{S' \to eS\}$ and $\{\}$
C. $\{S' \to \epsilon\}$ and $\{S' \to \epsilon\}$
D. $\{S' \to eS, S' \to \varepsilon\}$ and $\{S' \to \epsilon\}$

gate2003-cse compiler-design grammar normal

Answer

2.6.21 Grammar: GATE CSE 2003 | Question: 58 top

☛ https://gateoverflow.in/946

Consider the translation scheme shown below.

S → T R

R → + T $\{print('+');\}$ R$|\varepsilon$

T → **num** $\{print($**num**.val$);\}$

Here **num** is a token that represents an integer and **num**.val represents the corresponding integer value. For an input string '$9 + 5 + 2$', this translation scheme will print

A. $9 + 5 + 2$
B. $9\ 5 + 2+$
C. $9\ 5\ 2 + +$
D. $+ + 9\ 5\ 2$

gate2003-cse compiler-design grammar normal

Answer

2.6.22 Grammar: GATE CSE 2004 | Question: 45 top

☛ https://gateoverflow.in/1042

Consider the grammar with the following translation rules and E as the start symbol

$$\begin{array}{ll} E \to E_1 \# T & \{E.value = E_1.value * T.value\} \\ \quad |T & \{E.value = T.value\} \\ T \to T_1 \& F & \{T.value = T_1.value + F.value\} \\ \quad |F & \{T.value = F.value\} \\ F \to \text{num} & \{F.value = num.value\} \end{array}$$

Compute E.value for the root of the parse tree for the expression: 2 # 3 & 5 # 6 & 4

A. 200
B. 180
C. 160
D. 40

gate2004-cse compiler-design grammar normal

Answer

2.6.23 Grammar: GATE CSE 2004 | Question: 8 top

☛ https://gateoverflow.in/1005

Which of the following grammar rules violate the requirements of an operator grammar? P, Q, R are nonterminals, and r, s, t are terminals.

I. P → Q R

II. P → Q s R
III. P → ε
IV. P → Q t R r

A. (I) only
B. (I) and (III) only
C. (II) and (III) only
D. (III) and (IV) only

gate2004-cse compiler-design grammar normal

Answer

2.6.24 Grammar: GATE CSE 2004 | Question: 88 top

https://gateoverflow.in/1082

Consider the following grammar G:

$S \rightarrow bS \mid aA \mid b$

$A \rightarrow bA \mid aB$

$B \rightarrow bB \mid aS \mid a$

Let $N_a(w)$ and $N_b(w)$ denote the number of a's and b's in a string ω respectively.

The language $L(G)$ over $\{a, b\}^+$ generated by G is

A. $\{w \mid N_a(w) > 3N_b(w)\}$
B. $\{w \mid N_b(w) > 3N_a(w)\}$
C. $\{w \mid N_a(w) = 3k, k \in \{0, 1, 2, \dots\}\}$
D. $\{w \mid N_b(w) = 3k, k \in \{0, 1, 2, \dots\}\}$

gate2004-cse compiler-design grammar normal

Answer

2.6.25 Grammar: GATE CSE 2005 | Question: 59 top

https://gateoverflow.in/1382

Consider the grammar:

$$E \rightarrow E + n \mid E \times n \mid n$$

For a sentence $n + n \times n$, the handles in the right-sentential form of the reduction are:

A. $n, E + n$ and $E + n \times n$
B. $n, E + n$ and $E + E \times n$
C. $n, n + n$ and $n + n \times n$
D. $n, E + n$ and $E \times n$

gate2005-cse compiler-design grammar normal

Answer

2.6.26 Grammar: GATE CSE 2006 | Question: 32, ISRO2016-35 top

https://gateoverflow.in/995

Consider the following statements about the context free grammar

$$G = \{S \rightarrow SS, S \rightarrow ab, S \rightarrow ba, S \rightarrow \epsilon\}$$

I. G is ambiguous
II. G produces all strings with equal number of a's and b's
III. G can be accepted by a deterministic PDA.

Which combination below expresses all the true statements about G?

A. I only

B. I and III only
C. II and III only
D. I, II and III

gate2006-cse compiler-design grammar normal isro2016

Answer

2.6.27 Grammar: GATE CSE 2006 | Question: 59 top

https://gateoverflow.in/1837

Consider the following translation scheme.

- $S \to ER$
- $R \to *E\{\text{print}(`*');\}R \mid \varepsilon$
- $E \to F + E\{\text{print}(`+');\} \mid F$
- $F \to (S) \mid id\{\text{print}(id.value);\}$

Here id is a token that represents an integer and $id.value$ represents the corresponding integer value. For an input '$2 * 3 + 4$', this translation scheme prints

A. $2 * 3 + 4$
B. $2 * +3\ 4$
C. $2\ 3 * 4+$
D. $2\ 3\ 4 + *$

gate2006-cse compiler-design grammar normal

Answer

2.6.28 Grammar: GATE CSE 2006 | Question: 84 top

https://gateoverflow.in/1856

Which one of the following grammars generates the language $L = \{a^i b^j \mid i \neq j\}$?

A. $S \to AC \mid CB$
$C \to aCb \mid a \mid b$
$A \to aA \mid \varepsilon$
$B \to Bb \mid \varepsilon$

B. $S \to aS \mid Sb \mid a \mid b$

C. $S \to AC \mid CB$
$C \to aCb \mid \varepsilon$
$A \to aA \mid \varepsilon$
$B \to Bb \mid \varepsilon$

D. $S \to AC \mid CB$
$C \to aCb \mid \varepsilon$
$A \to aA \mid a$
$B \to Bb \mid b$

gate2006-cse compiler-design grammar normal theory-of-computation

Answer

2.6.29 Grammar: GATE CSE 2006 | Question: 85 top

https://gateoverflow.in/79801

The grammar

- $S \to AC \mid CB$
- $C \to aCb \mid \epsilon$
- $A \to aA \mid a$
- $B \to Bb \mid b$

generates the language $L = \{a^i b^j \mid i \neq j\}$. In this grammar what is the length of the derivation (number of steps starting from S) to generate the string $a^l b^m$ with $l \neq m$

A. $\max(l, m) + 2$
B. $l + m + 2$
C. $l + m + 3$
D. $\max(l, m) + 3$

gate2006-cse compiler-design grammar normal

Answer

2.6.30 Grammar: GATE CSE 2007 | Question: 52 top

https://gateoverflow.in/1250

Consider the grammar with non-terminals $N = \{S, C, S_1\}$, terminals $T = \{a, b, i, t, e\}$, with S as the start symbol, and the following set of rules:

$S \rightarrow iCtSS_1 \mid a$

$S_1 \rightarrow eS \mid \epsilon$

$C \rightarrow b$

The grammar is NOT LL(1) because:

A. it is left recursive
B. it is right recursive
C. it is ambiguous
D. it is not context-free

gate2007-cse compiler-design grammar normal

Answer

2.6.31 Grammar: GATE CSE 2007 | Question: 53 top

https://gateoverflow.in/1251

Consider the following two statements:

- P: Every regular grammar is LL(1)
- Q: Every regular set has a LR(1) grammar

Which of the following is **TRUE**?

A. Both P and Q are true
B. P is true and Q is false
C. P is false and Q is true
D. Both P and Q are false

gate2007-cse compiler-design grammar normal

Answer

2.6.32 Grammar: GATE CSE 2007 | Question: 78 top

https://gateoverflow.in/1272

Consider the CFG with $\{S, A, B\}$ as the non-terminal alphabet, $\{a, b\}$ as the terminal alphabet, S as the start symbol and the following set of production rules:

$S \rightarrow aB \quad\quad S \rightarrow bA$

$B \rightarrow b \quad\quad A \rightarrow a$

$B \rightarrow bS \quad\quad A \rightarrow aS$

$B \rightarrow aBB \quad\quad S \rightarrow bAA$

Which of the following strings is generated by the grammar?

A. $aaaabb$
B. $aabbbb$
C. $aabbab$
D. $abbbba$

gate2007-cse compiler-design grammar normal

Answer

2.6.33 Grammar: GATE CSE 2007 | Question: 79 top

☛ https://gateoverflow.in/43512

Consider the CFG with $\{S, A, B\}$ as the non-terminal alphabet, $\{a, b\}$ as the terminal alphabet, S as the start symbol and the following set of production rules:

$S \rightarrow aB$ $\quad S \rightarrow bA$

$B \rightarrow b$ $\quad A \rightarrow a$

$B \rightarrow bS$ $\quad A \rightarrow aS$

$B \rightarrow aBB$ $\quad S \rightarrow bAA$

For the string $aabbab$, how many derivation trees are there?

A. 1
B. 2
C. 3
D. 4

gate2007-cse compiler-design grammar normal

Answer

2.6.34 Grammar: GATE CSE 2008 | Question: 50 top

☛ https://gateoverflow.in/395

Which of the following statements are true?

I. Every left-recursive grammar can be converted to a right-recursive grammar and vice-versa
II. All ϵ-productions can be removed from any context-free grammar by suitable transformations
III. The language generated by a context-free grammar all of whose productions are of the form $X \rightarrow w$ or $X \rightarrow wY$ (where, w is a string of terminals and Y is a non-terminal), is always regular
IV. The derivation trees of strings generated by a context-free grammar in Chomsky Normal Form are always binary trees

A. I, II, III and IV
B. II, III and IV only
C. I, III and IV only
D. I, II and IV only

gate2008-cse normal compiler-design grammar

Answer

2.6.35 Grammar: GATE CSE 2010 | Question: 38 top

☛ https://gateoverflow.in/2339

The grammar $S \rightarrow aSa \mid bS \mid c$ is

A. LL(1) but not LR(1)
B. LR(1) but not LL(1)
C. Both LL(1) and LR(1)
D. Neither LL(1) nor LR(1)

gate2010-cse compiler-design grammar normal

Answer

2.6.36 Grammar: GATE CSE 2014 Set 2 | Question: 17 top

☛ https://gateoverflow.in/1973

Consider the grammar defined by the following production rules, with two operators $*$ and $+$

- $S \rightarrow T * P$
- $T \rightarrow U \mid T * U$
- $P \rightarrow Q + P \mid Q$
- $Q \rightarrow Id$
- $U \rightarrow Id$

Which one of the following is TRUE?

A. $+$ is left associative, while $*$ is right associative
B. $+$ is right associative, while $*$ is left associative
C. Both $+$ and $*$ are right associative
D. Both $+$ and $*$ are left associative

gate2014-cse-set2 compiler-design grammar normal

Answer

2.6.37 Grammar: GATE CSE 2016 Set 2 | Question: 45 top

https://gateoverflow.in/39594

Which one of the following grammars is free from left recursion?

A. $S \rightarrow AB$
$A \rightarrow Aa \mid b$
$B \rightarrow c$
B. $S \rightarrow Ab \mid Bb \mid c$
$A \rightarrow Bd \mid \epsilon$
$B \rightarrow e$
C. $S \rightarrow Aa \mid B$
$A \rightarrow Bb \mid Sc \mid \epsilon$
$B \rightarrow d$
D. $S \rightarrow Aa \mid Bb \mid c$
$A \rightarrow Bd \mid \epsilon$
$B \rightarrow Ae \mid \epsilon$

gate2016-cse-set2 compiler-design grammar easy

Answer

2.6.38 Grammar: GATE CSE 2016 Set 2 | Question: 46 top

https://gateoverflow.in/39598

A student wrote two context-free grammars G1 and G2 for generating a single C-like array declaration. The dimension of the array is at least one. For example,

```
int a[10] [3];
```

The grammars use D as the start symbol, and use six terminal symbols **int ; id [ ] num.**

Grammar **G1**	Grammar **G2**
D $\rightarrow$ **int L;**	D $\rightarrow$ **int L;**
L $\rightarrow$ **id** [**E**	L $\rightarrow$ **id E**
E $\rightarrow$ **num**]	E $\rightarrow$ **E** [**num**]
E $\rightarrow$ **num**] [**E**	E $\rightarrow$ [**num**]

Which of the grammars correctly generate the declaration mentioned above?

A. Both **G1** and **G2**
B. Only **G1**
C. Only **G2**
D. Neither **G1** nor **G2**

gate2016-cse-set2 compiler-design grammar normal

Answer

2.6.39 Grammar: GATE CSE 2017 Set 2 | Question: 32 top

https://gateoverflow.in/118374

Consider the following expression grammar G:

- $E \rightarrow E - T \mid T$
- $T \rightarrow T + F \mid F$
- $F \rightarrow (E) \mid id$

Which of the following grammars is not left recursive, but is equivalent to G?

A. $E \to E - T \mid T$
$T \to T + F \mid F$
$F \to (E) \mid id$

B. $E \to TE'$
$E' \to -TE' \mid \epsilon$
$T \to T + F \mid F$
$F \to (E) \mid id$

C. $E \to TX$
$X \to -TX \mid \epsilon$
$T \to FY$
$Y \to +FY \mid \epsilon$
$F \to (E) \mid id$

D. $E \to TX \mid (TX)$
$X \to -TX \mid +TX \mid \epsilon$
$T \to id$

gate2017-cse-set2 grammar

Answer

2.6.40 Grammar: GATE CSE 2019 | Question: 43 top

☛ https://gateoverflow.in/302805

Consider the augmented grammar given below:

- $S' \to S$
- $S \to \langle L \rangle \mid id$
- $L \to L, S \mid S$

Let $I_0 = \text{CLOSURE}(\{[S' \to \cdot S]\})$. The number of items in the set $\text{GOTO}(I_0, \langle\ )$ is______

gate2019-cse numerical-answers compiler-design grammar

Answer

2.6.41 Grammar: GATE CSE 2021 Set 1 | Question: 31 top

☛ https://gateoverflow.in/357420

Consider the following context-free grammar where the set of terminals is $\{a, b, c, d, f\}$.

$$\begin{aligned} \text{S} &\to d\,a\,\text{T} \mid \text{R}\,f \\ \text{T} &\to a\,\text{S} \mid b\,a\,\text{T} \mid \epsilon \\ \text{R} &\to c\,a\,\text{T}\,\text{R} \mid \epsilon \end{aligned}$$

The following is a partially-filled LL(1) parsing table.

	a	b	c	d	f	\$
S			$\boxed{1}$	$\text{S} \to da\text{T}$	$\boxed{2}$	
T	$\text{T} \to a\text{S}$	$\text{T} \to ba\text{T}$	$\boxed{3}$		$\text{T} \to \varepsilon$	$\boxed{4}$
R			$\text{R} \to ca\text{TR}$		$\text{R} \to \varepsilon$	

Which one of the following choices represents the correct combination for the numbered cells in the parsing table ("blank" denotes that the corresponding cell is empty)?

A. $\boxed{1}$ $\text{S} \to \text{R}f$ $\boxed{2}$ $\text{S} \to \text{R}f$ $\boxed{3}$ $\text{T} \to \varepsilon$ $\boxed{4}$ $\text{T} \to \varepsilon$
B. $\boxed{1}$ blank $\boxed{2}$ $\text{S} \to \text{R}f$ $\boxed{3}$ $\text{T} \to \varepsilon$ $\boxed{4}$ $\text{T} \to \varepsilon$
C. $\boxed{1}$ $\text{S} \to \text{R}f$ $\boxed{2}$ blank $\boxed{3}$ blank $\boxed{4}$ $\text{T} \to \varepsilon$
D. $\boxed{1}$ blank $\boxed{2}$ $\text{S} \to \text{R}f$ $\boxed{3}$ blank $\boxed{4}$ blank

gate2021-cse-set1 compiler-design grammar

Answer

2.6.42 Grammar: GATE IT 2007 | Question: 9 top

https://gateoverflow.in/3442

Consider an ambiguous grammar G and its disambiguated version D. Let the language recognized by the two grammars be denoted by $L(G)$ and $L(D)$ respectively. Which one of the following is true?

A. $L(D) \subset L(G)$
B. $L(D) \supset L(G)$
C. $L(D) = L(G)$
D. $L(D)$ is empty

gate2007-it compiler-design grammar normal

Answer

Answers: Grammar

2.6.1 Grammar: GATE CSE 1990 | Question: 16a top

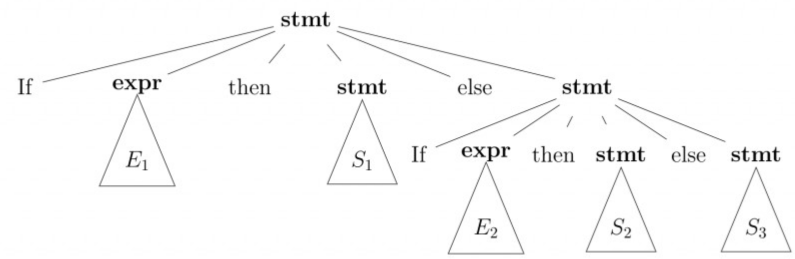
https://gateoverflow.in/86869

✔ The given grammar is well known as "Dangling Else" problem. The given grammar is ambiguous and ambiguity can be resolved.

$$\begin{aligned}\textbf{stmt} \rightarrow\ & \textbf{if}\ \text{expr}\ \textbf{then}\ \text{stmt}\\ & \mid \textbf{if}\ \text{expr}\ \textbf{then}\ \text{stmt}\ \textbf{else}\ \text{stmt}\\ & \mid \textbf{other}\end{aligned}$$

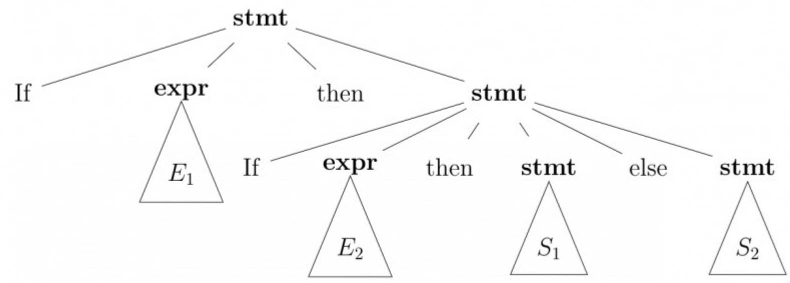
Consider the compound conditional statement for the above grammar

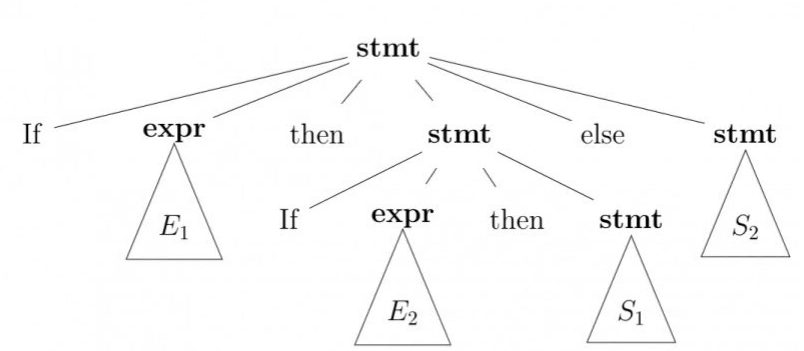
if E_1 **then** S_1 **else** if E_2 **then** S_2 **else** S_3 has the following parse tree



Well this is ambiguous due to the statement
if E_1 **then if** E_2 **then** S_1 **else** S_2

The two parse trees are





Ambiguity can be resolved by parsers as follows

> In all programming languages with conditional statements of this form, the first parse tree is preferred. The general rule is, "Match each else with the closest unmatched then"

In practice it is rarely built into the productions .

stmt $\rightarrow$ **matched_stmt**
| **open_stmt**
matched_stmt $\rightarrow$ **if** expr **then** matched_stmt **else** matched_stmt
| **other**
open_stmt $\rightarrow$ **if** expr **then** stmt
| **if** expr **then** matched_stmt **else** open_stmt

However this grammar is also ambiguous

Consider the following exercise question from Dragon book.

Exercise 4.3.3: The following grammar is proposed to remove the "dangling-else ambiguity":

- $stmt \rightarrow$ **if** $expr$ **then** $stmt$
  $|matchedStmt$
- $matchedStmt \rightarrow$ **if** $expr$ **then** $matchedStmt$ **else** $stmt$
  | **other**

Show that this grammar is still ambiguous.

Solution is given here. But we can make an equivalent unambiguous grammar for the dangling-else problem and that makes the language not inherently ambiguous.

Try running this code in any compilers . No doubt that all compilers will successfully parse and produce output

```
#include <stdio.h>

int main(void) {
 if(1<3)
 if(1<2)
 printf("1 is the smallest");
 else
 printf("2 is the smallest");
 return 0;
}
```

Thus we can say that ambiguity in Dangling Else Problem can be resolved and we can have an unambiguous grammar for it (making the language **NOT** inherently ambiguous. But many of the grammars given for Dangling Else problem is ambiguous.

Source

- Compilers: Principles, Techniques, & Tools -Aho & Ullman

Good Read

- http://cs.stackexchange.com/questions/32475/resolving-ambiguity-in-dangling-else
- https://gateoverflow.in/54718/inherently-ambiguous-languages-deterministic-context-grammars
- https://gateoverflow.in/84451/dangling-else-problem-and-ambiguity-elimination

References

👍 20 votes -- Akhil Nadh PC (16.5k points)

2.6.2 Grammar: GATE CSE 1991 | Question: 10a top

☛ https://gateoverflow.in/537

✔ Yes. It is unambiguous grammar since for any string no more than 1 parse tree is possible.

For precedency draw the parse tree and find the depth of operator - and / .

Here "/" having more depth than " - " operator so precedency of" /" is higher than "-".

👍 17 votes -- papesh (18k points)

2.6.3 Grammar: GATE CSE 1991 | Question: 10b top

☛ https://gateoverflow.in/43604

If the expression with redundant parentheses is (id/id) or $id-(id/id)$ then it can be generated by the given grammar.

To generate expression (id/id) we can go through following steps-

1. $E \to T$
2. $E \to F \qquad (T \to F)$
3. $E \to (E) \qquad (F \to (E))$
4. $E \to (T) \qquad (E \to T)$
5. $E \to (T/F) \qquad (T \to T/F)$
6. $E \to (F/F) \qquad (T \to F)$
7. $E \to (id/id) \qquad (F -> id)$

Now to generate expression $id-(id/id)$ we can go through following steps-

1. $E \to E - T$
2. $E \to E - F \qquad (T \to F)$
3. $E \to E - (E) \qquad (F \to (E))$
4. $E \to E - (T) \qquad (E \to T)$
5. $E \to E - (T/F) \qquad (T \to T/F)$
6. $E \to E - (F/F) \qquad (T \to F)$
7. $E \to T - (F/F) \qquad (E \to T)$
8. $E \to F - (F/F) \qquad (T \to F)$
9. $E \to id - (id/id) \qquad (F \to id)$

👍 6 votes -- Dhananjay Kumar Sharma (18.8k points)

2.6.4 Grammar: GATE CSE 1991 | Question: 10c top

☛ https://gateoverflow.in/43605

✔ Here we have to convert the grammar into one which does not generate expressions with redundant parentheses.So the grammar which does not generate expressions with redundant parentheses

- $E \to E - T \mid T$
- $E' \to E - T$
- $T \to id \mid T'/F$
- $T' \to F \mid T'/F$
- $F \to id \mid (E')$

Now equivalent grammar after removing left recursion

- $E \to TX'$
- $X' \to -TX' \mid \epsilon$
- $E' \to E - T$
- $T \to id \mid T'/F$
- $T' \to FY'$

- $Y' \to /FY' \mid \epsilon$
- $F \to id \mid (E')$

18 votes -- Manoj Kumar (26.7k points)

2.6.5 Grammar: GATE CSE 1994 | Question: 1.18 top https://gateoverflow.in/2461

✔ Correct Option: **C**

Since CFG's are used to show syntactic rules while designing compiler, and syntactic rules don't check for meaningful things such as if a variable has been declared before its use or not. Such things are meant to be handled by Semantic Analysis phase (requires power of a context sensitive grammar).

For D, a regular expression does not restrict the string length. Languages have restriction for variable name length for storing purpose like in symbol table.

For A, if then else is inherently ambiguous. But CFG can represent inherently ambiguous languages just that there are more than one parse trees possible for some strings.

49 votes -- Gate Keeda (15.9k points)

2.6.6 Grammar: GATE CSE 1994 | Question: 20 top https://gateoverflow.in/2516

✔ Its answer is $2n-1$ for n length string, because in CNF at every step only 1 terminal can replace a variable, for example

- $S \to AB$
- $A \to a$
- $B \to c$

For generating string 'ac' 3 productions will be used.

Reference:- Peter Linz

31 votes -- Manu Thakur (34.1k points)

2.6.7 Grammar: GATE CSE 1994 | Question: 3.5 top https://gateoverflow.in/2482

✔ Backus normal form (BNF) is a notation technique for context-free grammars, often used to describe the syntax of languages used in computing
Yacc (Yet Another Compiler-Compiler) is a computer program for the Unix operating system. It is a Look Ahead Left-to-Right (LALR) parser generator, generating a parser, the part of a compiler that tries to make syntactic sense of the source code, specifically a LALR parser, based on an analytic grammar
Yacc is written in portable C. The class of specifications accepted is a very general one: LALR(1) grammars with disambiguating rules.

17 votes -- set2018 (6.4k points)

2.6.8 Grammar: GATE CSE 1995 | Question: 1.10 top https://gateoverflow.in/2597

✔

- $S\alpha \to$

This violates the condition of context-free grammar that the LHS must be a single non-terminal symbol.

- $S\alpha \to b$

This violates even the weaker requirement for CSG that the length of RHS of a production must be at least same as that of LHS. So, the grammar is not even context-sensitive.

Ref: https://stackoverflow.com/questions/8236422/context-free-grammars-versus-context-sensitive-grammars

References

👍 44 votes -- Arjun Suresh (330k points)

2.6.9 Grammar: GATE CSE 1995 | Question: 9 top

☛ https://gateoverflow.in/2644

A grammer having left recursion generates a cycle.

And no left recursive grammer is LL(1) grammer.

👍 15 votes -- simran (103 points)

2.6.10 Grammar: GATE CSE 1996 | Question: 11 top

☛ https://gateoverflow.in/2763

✔ Final grammar is

- $S \to ABAC \mid ABC \mid BAC \mid BC \mid AC \mid AAC \mid d$
- $A \to aA \mid a$
- $B \to bB \mid b$
- $C \to d$

👍 46 votes -- Mithlesh Upadhyay (4.3k points)

2.6.11 Grammar: GATE CSE 1996 | Question: 2.10 top

☛ https://gateoverflow.in/2739

✔ (d) the sentence

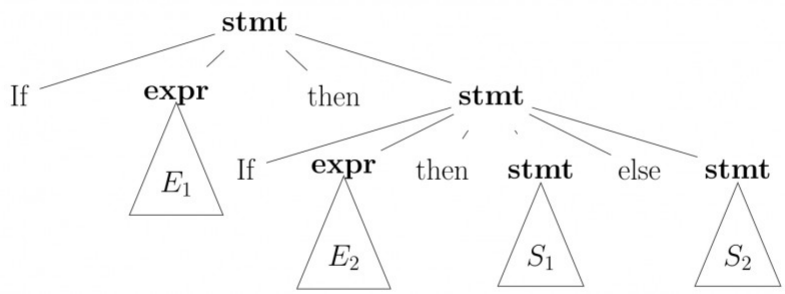
- if a then if b then c: = d else c:= f `has two parse trees as follows:`

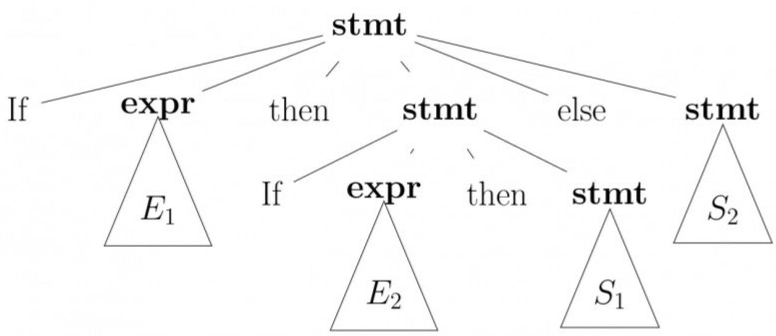
1. if a then (if b then c:= d) else c:= f
2. and if a then (if b then c:=d else c:= f)

Ambiguity - "dangling else"

```
stmt -> if expr then stmt |
        if expr then stmt else stmt | other stmts
```

if E_1 **then** **if** E_2 **then** S_1 **else** S_2





👍 32 votes -- Arjun Suresh (330k points)

2.6.12 Grammar: GATE CSE 1997 | Question: 1.6 top

☛ https://gateoverflow.in/2222

✔ It will be A. For multiple '$\oplus$', the derivation is possible only via 'X' which is on left side of '$\oplus$' in the production. Hence it is left associative.

For multiple '$*$', the derivation is possible only via 'Y' which is on the right side of '$*$' in the production. Hence it is right associative.

If both left and right derivations were possible, the grammar would have been ambiguous and we couldn't have given associativity.

👍 37 votes -- Gate Keeda (15.9k points)

2.6.13 Grammar: GATE CSE 1997 | Question: 11 top

☛ https://gateoverflow.in/2271

✔

a. $i+k=j+m$
   where $i, j, k, m >= 0$

b. $bcde$

👍 17 votes -- Danish (3.4k points)

2.6.14 Grammar: GATE CSE 1998 | Question: 14 top

☛ https://gateoverflow.in/1728

✔ (a) $L(G_1) = \{a^n b^m \mid n \neq m\}$

(b)

- $S_1 \rightarrow S_2S_3 \mid S_3S_2 \qquad (2)$
- $S_3 \rightarrow aS_3b \mid ab \qquad (2)$
- $S_2 \rightarrow ab \qquad (1)$

So, totally $2+2+1=5$ extra productions.

(c) If grammar is ambiguous and no unambiguous grammar is possible for the language, then language is inherently ambiguous.

Non-determinism of language (if it means, L is not DCFL) + ambiguity for all possible grammars implies inherent ambiguity. Or if a language is deterministic, it is surely not inherently ambiguous. But if a language is not deterministic, it may or may not be inherently ambiguous.

The given language L_2 is the set of strings where number of $a's$ is followed by an equal number of $b's$ and if not, for the remaining part, the number of $a's$ is followed by an equal number of $b's$. This is actually a deterministic language as we do not need to do multiple counts here. Only after the first condition is violated we need to start doing the count for the second part. This makes the language deterministic and hence it cannot be inherently ambiguous.

👍 13 votes -- Praveen Saini (41.9k points)

2.6.15 Grammar: GATE CSE 1998 | Question: 6b top

☛ https://gateoverflow.in/1697

✔ (b) As it is the case of indirect recursion so let first make it as direct recursion then apply rules of removal of left recursion.

to make it as direct recursion first production remain unchanged while in second production substitute the right hand side of first production wherever it comes.In the question S comes in middle of A so substitute the right hand side of production S.Now after substituting it looks like:

- $A \rightarrow Ac \mid Aad \mid bd \mid \epsilon$

Now remove direct recursion from it

For removal of direct recursion rule:

- $A \rightarrow A\alpha_1 \mid \ldots \mid A\alpha_n \mid \beta_1 \mid \ldots \mid \beta_m$

Replace these with two sets of productions, one set for A :

- $A \rightarrow \beta_1 A' \mid \ldots \mid \beta_m A'$

and another set for the fresh nonterminal A'

- $A' \to \alpha_1 A' \mid \ldots \mid \alpha_n A' \mid \epsilon$

After applying these rule we'll get:

- $A \to bdA' \mid A'$
- $A' \to cA' \mid adA' \mid \epsilon$

Now complete production without left recursion is:

- $S \to Aa \mid b$
- $A \to bdA' \mid A'$
- $A' \to cA' \mid adA' \mid \epsilon$

32 votes -- shashi shekhar (437 points)

2.6.16 Grammar: GATE CSE 1999 | Question: 2.15 top

https://gateoverflow.in/1493

✔ Let grammar is like this :
$S \to a$
$A \to AbA$
This grammar is left as well as right recursive but still unambiguous.. **A** is useless production but still part of grammar..
A grammar having both left as well as right recursion may or may not be ambiguous ..

Let's take a grammar say

$S \to SS$

Now, according to the link https://en.wikipedia.org/wiki/Formal_grammar

> For a grammar G, if we have L(G) then all the strings/sentences in this language can be produced after some finite number of steps .

But, for the grammar

$$S \to SS$$

Can we produce any string after a finite number of steps ? NO, and hence language of this grammar is empty set {} .

Hence, If a grammar is having both left and right recursion, then grammar may or may not be ambiguous .

Correct Answer: C

References

52 votes -- Digvijay (44.9k points)

2.6.17 Grammar: GATE CSE 2000 | Question: 2.21, ISRO2015-24 top

https://gateoverflow.in/668

✔ Correct Option: **B**

Operator which is at lower level in the grammar is termed to have higher precedence.

34 votes -- Gate Keeda (15.9k points)

2.6.18 Grammar: GATE CSE 2001 | Question: 1.18 top

https://gateoverflow.in/711

✔ Correct Option: **A**
(A) We can not have different Left Most Derivation and Right Most Derivation parse trees BUT we can certainly have different LMD and RMD for a given string.(LMD and RMD here refers to the order of various productions used for derivation which could be different.)

(D) is wrong w.r.t. question because IT IS TRUE that any LR(k) IS NEVER AMBIGUOUS and so an ambiguous can never be an LR(K) for any k, no matter how large k becomes.

(B) and (C) can not be the answer because LL(1) is TOP-DOWN parser and LALR is powerful than SLR. So both are TRUE.

👍 42 votes -- Sandeep_Uniyal (6.5k points)

2.6.19 Grammar: GATE CSE 2001 | Question: 18 top ☛ https://gateoverflow.in/759

✔

a. $S \rightarrow aS' \mid bS'$
$S' \rightarrow aS' \mid bS' \mid \epsilon$

b. $S \rightarrow aSbS \rightarrow abS \rightarrow abaSbS \rightarrow ababS \rightarrow abab$
$S \rightarrow aSbS \rightarrow abSaSbS \rightarrow abaSbS \rightarrow ababS \rightarrow abab$

The above two derivations corresponds to two different parse trees and hence the grammar is ambiguous.

👍 30 votes -- Pooja Palod (24.1k points)

2.6.20 Grammar: GATE CSE 2003 | Question: 56 top ☛ https://gateoverflow.in/944

✔

- $\text{FIRST}(S) = \{i, a\}$
- $\text{FIRST}(S') = \{e, \epsilon\}$
- $\text{FIRST}(E) = \{b\}$
- $\text{FOLLOW}(S') = \{e, \$\}$

Only when FIRST contains ϵ, we need to consider FOLLOW for getting the parsing table entry.

$M[S', e] = \{S' \rightarrow eS(\text{FIRST}), S' \rightarrow \epsilon \text{ (considering FOLLOW)}\}$

$M[S', \$] = \{S \rightarrow \epsilon\}$

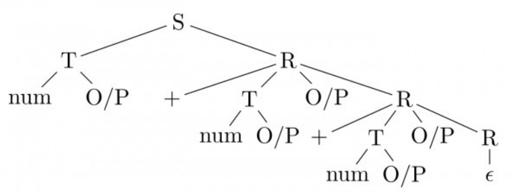
	a	i	b	e	t	$\$$
S	$S \rightarrow a$	$S \rightarrow ietSS'$				
S'				$S' \rightarrow eS$, $S' \rightarrow \epsilon$		$S' \rightarrow \epsilon$
E			$E \rightarrow b$			

Answer is **D**

👍 50 votes -- Pooja Palod (24.1k points)

2.6.21 Grammar: GATE CSE 2003 | Question: 58 top ☛ https://gateoverflow.in/946

✔ Correct Option: **B**
$9\ 5 + 2+$



👍 45 votes -- Amar Vashishth (25.2k points)

2.6.22 Grammar: GATE CSE 2004 | Question: 45 top ☛ https://gateoverflow.in/1042

✔ Here # is multiplication and & is addition by semantics rules given in the question.
By observation of productions,

1. here &(+) is higher precedence than #(*), because & is far from starting symbol
2. both &,# are left associative

So, we can solve the expression as $((2*(3+5))*(6+4)) = 160$

Answer is (**C**).

44 votes -- minal (13.1k points)

2.6.23 Grammar: GATE CSE 2004 | Question: 8 top https://gateoverflow.in/1005

answer is **B**.

Operator grammar cannot contain

1. Nullable variable
2. Two adjacent non-terminal on RHS of production

35 votes -- koushiksngh264 (191 points)

2.6.24 Grammar: GATE CSE 2004 | Question: 88 top https://gateoverflow.in/1082

above CFG generate string b, aaa..
b will eliminate options A and D
aaa eliminate options B.
C is answer i.e. number of $a = 3k, k = 0, 1, 2 \ldots$.

32 votes -- Digvijay (44.9k points)

2.6.25 Grammar: GATE CSE 2005 | Question: 59 top https://gateoverflow.in/1382

- $n + n * n$
- $E + n * n$
- $E * n$
- E

String in RED indicates handle here

So, answer is D

37 votes -- Pooja Palod (24.1k points)

2.6.26 Grammar: GATE CSE 2006 | Question: 32, ISRO2016-35 top https://gateoverflow.in/995

I. **True.** G is ambiguous. E.g. the string ab has multiple derivation trees like $S \to SS \to abS \to ab$, and $S \to ab$.

II. **False.** G does not produce all strings with equal no. of a\`s and $b$\`s. ($aabb$ cannot be generated).

III. **True.** The given grammar G generates the language $(ab+ba)^*$, which is Regular and therefore also DCFL. So, a D-PDA can be designed for G.

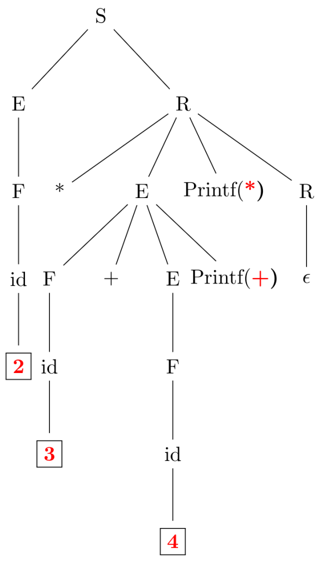
Hence, the answer is option **B**.

106 votes -- Pooja Palod (24.1k points)

2.6.27 Grammar: GATE CSE 2006 | Question: 59 top https://gateoverflow.in/1837

Correct Option: **D**

Make a tree and perform post order evaluation.



37 votes

-- Gate Keeda (15.9k points)

2.6.28 Grammar: GATE CSE 2006 | Question: 84 top

https://gateoverflow.in/1856

✔ **option A**

C$\Rightarrow$ a

or, C $\Rightarrow$b

or, C $\Rightarrow$aCb$\Rightarrow$aaCbb$\Rightarrow$ aaaCbbb .. soon

at last you have to put either C$\rightarrow$ a or C$\rightarrow$ b

so production C is used to derive $a^{n+1}b^n$ or a^nb^{n+1} $n \geq 0$

S$\rightarrow$ AC [Aa^{n}b^{n+1}] can make $a^{n+1}b^{n+1}$ as single a can be derived from A [A $\Rightarrow$aA $\Rightarrow$a as A$\rightarrow$ ϵ] , similarly S$\rightarrow$ CB

simple way, ab can be derived from grammar as S $\Rightarrow$ AC $\Rightarrow$aAC $\Rightarrow$aC$\Rightarrow$ ab

option A is wrong

option B , language is used to drive a^+b^* or a^*b^+ , ab will be derived as S$\Rightarrow$aS $\Rightarrow$ab

option B is wrong

Option C

C $\Rightarrow$ ϵ

or C $\Rightarrow$ aCb $\Rightarrow$ aaCbb $\Rightarrow$ aaaCbbb .. soon at last need to put C $\rightarrow \epsilon$

Production C will generate a^nb^n $n \geq 0$

S$\rightarrow$ AC can generate a^nb^n as A can be ϵ , similarly S $\rightarrow$CB

option C is wrong

Option D .

production C is used for generate a^nb^n as in option C

S$\rightarrow$ AC will increase no of a's before a^nb^n

as A will generate a or aa or aaa... i,e a^+ , so S$\rightarrow$ AC will generate $a^+a^nb^n$, i.e a^ib^j $i > j$

S $\rightarrow$ CB will generate $a^nb^nb^+$ i.e a^ib^j $i < j$

option D is right .

45 votes

-- Praveen Saini (41.9k points)

2.6.29 Grammar: GATE CSE 2006 | Question: 85 top

https://gateoverflow.in/79801

✔ $L = a^lb^m$ $l \neq m$ means either $l > m$ or $l < m$

Case I [l > m]

if $l > m$,a^lb^m can be written as $\mathbf{a^{l-m}a^mb^m}$ [$l - m$ cannot be 0 as l should be $> m$]

S$\rightarrow$ AC , one step

a^{l-m} use $l-m$ steps using productions of A
[as $l-m=1$, one step A $\to$ a
$l-m=2$, two steps A $\to$aA $\to$aa
$l-m=3$, three steps , A $\to$aA$\to$aaA$\to$aaa.. so on]
a^mb^m will be generate in $m+1$ steps using production C
[as $m=0$ one step C $\to \epsilon$
$m=1$, two steps C $\to$ aCb $\to$ ab
$m=$ 2, three steps C $\to$ aCb $\to$aaCbb$\to$ aabb.. so on]
so if $l>m$ total steps $= 1+l-m+m+1 = l+2$

Case II [l<m]

simillar if $l<m$, a^lb^m can be written as $\mathbf{a^lb^lb^{m-l}}$ [$m-l$ cannot be 0 as m should be $>l$]
S$\to$ CB one step
a^lb^l will be derives using $l+1$ steps
b^{m-l} will be derived using $m-l$ steps
total $= 1+l+1+m-l = m+2$
so $L = a^lb^m \ l \neq m$ will take $max(l,m)+2$ steps
Correct Answer: A

16 votes -- Praveen Saini (41.9k points)

Option **A** is correct. We can try some strings:

$w = a^lb^m, l \neq m \Rightarrow aabbbS \to \underset{1}{CB} \to \underset{2}{aCbB} \to \underset{3}{aaCbbB} \to \underset{4}{aaCbbB} \to \underset{5}{aabbb}$

$w = aaaabbS \to \underset{1}{AC} \to \underset{2}{AaCb} \to \underset{3}{AaaCbb} \to \underset{4}{Aaa\epsilon bb} \to \underset{5}{aAaabb} \to \underset{6}{aaaabb}$

$w = aaaab$
$S \to \underset{1}{AC} \to \underset{2}{AaCb} \to \underset{3}{Aa\varepsilon b} \to \underset{4}{aAab} \to \underset{5}{aaAab} \to \underset{6}{aaaab}$

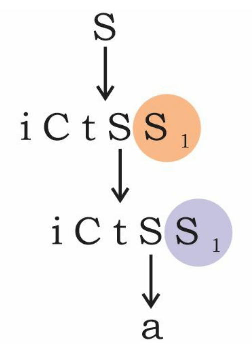
In cases 2 and 3 reducing number of $b's$ has no affect on the number of productions required.

20 votes -- rameshbabu (2.6k points)

2.6.30 Grammar: GATE CSE 2007 | Question: 52 top

https://gateoverflow.in/1250

✔ Here, we can expand any one of S_1 to $\in$ and other to ea, but which one will it be need not matter, because in the end we will still get the same string.



This means that the Grammar is Ambiguous. LL(1) cannot be ambiguous. Here's a Proof for that

LL(1) Derivations

Left to Right Scan of input
Leftmost Derivation
(1) look ahead 1 token at each step
Alternative characterization of LL(1) Grammars:

Whenever $A \to \alpha \mid \beta \in G$

1. $\text{FIRST}(\alpha) \cap \text{FIRST}(\beta) = \{\}$, and
2. If $\alpha \overset{*}{\Longrightarrow} \varepsilon$ then $\text{FIRST}(\beta) \cap \text{FOLLOW}(A) = \{\}$.

Corollary: No Ambiguous Grammar is $\text{LL}(1)$.

Correct Answer: C.

References

46 votes -- Amar Vashishth (25.2k points)

2.6.31 Grammar: GATE CSE 2007 | Question: 53 top https://gateoverflow.in/1251

✔ Answer: **option C**

LL Grammar: Grammars which can be parsed by an LL parser.

LL parser: Parses the input from **L**eft to right, and constructs a **L**eftmost derivation of the sentence(i.e. it is always the leftmost non-terminal which is rewritten). LL parser is a top-down parser for a subset of context-free languages.
An LL parser is called an LL(k) parser if it uses k tokens of lookahead when parsing a sentence and can do it without backtracking.

Consider a Grammar G:

- $S \to a \mid aa$

This grammar is **Regular** but cannot be parsed by a LL(1) parser w/o backtracking, because here, lookahead is of 1 symbol only and in the grammar for both productions, parser while looking at just one(first) symbol, which is a, fails to select the correct rule for parsing.

Hence, not every Regular grammar is LL(1); Statement P is False.

LR Grammar: Grammars which can be parsed by LR parsers.

LR Parser: They are a type of bottom-up parsers that efficiently handle deterministic context-free languages(DCFL) in guaranteed linear time.

All Regular Languages are also DCFL. Hence, they all can be parsed by a LR(1) grammar.

Hence, Statement Q is True.

78 votes -- Amar Vashishth (25.2k points)

2.6.32 Grammar: GATE CSE 2007 | Question: 78 top https://gateoverflow.in/1272

✔ $S \to aB$
$\to aaBB$
$\to aabB$
$\to aabbS$
$\to aabbaB$
$\to aabbab$

Correct Answer: C

27 votes -- Arjun Suresh (330k points)

2.6.33 Grammar: GATE CSE 2007 | Question: 79 top https://gateoverflow.in/43512

✔ $S \to aB$
$\to aaBB$
$\to aabB$
$\to aabbS$
$\to aabbaB$
$\to aabbab$

$S \to aB$

$\rightarrow aaBB$ (till now, only 1 choice possible)
$\rightarrow aabSB$ (last time we took $B \rightarrow b$, now taking $B \rightarrow bS$)
$\rightarrow aabbAB$
$\rightarrow aabbaB$
$\rightarrow aabbab$

So, totally 2 possible derivation trees.

Correct Answer: B

31 votes -- Arjun Suresh (330k points)

2.6.34 Grammar: GATE CSE 2008 | Question: 50 top https://gateoverflow.in/395

Answer is C:

Statement 1 is **true**: Using GNF we can convert Left recursive grammar to right recursive and by using reversal of CFG and GNF we can convert right recursive to left recursive.

Statement 2 is **false**: because if ϵ is in the language then we can't remove ϵ production from Start symbol. (For example $L = a^*$)

Statement 3 is **true** because right linear grammar generates regular set

Statement 4 is **true**, only two non-terminals are there in each production in CNF. So it always form a binary tree.

59 votes -- Vikrant Singh (11.2k points)

2.6.35 Grammar: GATE CSE 2010 | Question: 38 top https://gateoverflow.in/2339

Correct Option: **C**

For LL(1) take First(S). and do intersection between the result. if intersection is Phi then LL(1) else not.

Making a parsing table and checking if there are two or more entries under any terminal. If yes then neither LL(1) nor LR(1).

42 votes -- Gate Keeda (15.9k points)

2.6.36 Grammar: GATE CSE 2014 Set 2 | Question: 17 top https://gateoverflow.in/1973

$P \rightarrow Q + P$ here P is to right of $+$

So, $+$ is right associative

Similarly, for $T \rightarrow T * U$ $*$ is left associative as T is to left of $*$

So, answer is **B**

31 votes -- Pooja Palod (24.1k points)

2.6.37 Grammar: GATE CSE 2016 Set 2 | Question: 45 top https://gateoverflow.in/39594

Option (A) has immediate left recursion." $A \rightarrow Aa$"

Option (C) has indirect left recursion "$S \rightarrow Aa \overset{A \rightarrow Sc}{\Longrightarrow} Sca$"

Option (D) has indirect left recursion "$A \rightarrow Bd \overset{B \rightarrow Ae}{\Longrightarrow} Aed$"

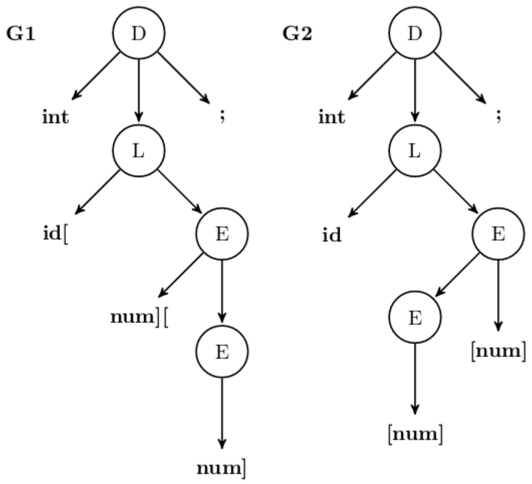
Option (B) is free from left recursion. No direct left recursion. No indirect left recursion.

Correct Option: **B**

51 votes -- Ashish Deshmukh (1.3k points)

2.6.38 Grammar: GATE CSE 2016 Set 2 | Question: 46 top https://gateoverflow.in/39598

Correct Option: **A** (Both $G1$ and $G2$)



49 votes

-- Shashank Chavan (2.4k points)

2.6.39 Grammar: GATE CSE 2017 Set 2 | Question: 32 top

https://gateoverflow.in/118374

✔ Since, the grammar given in the question is left recursive, we need to remove left recursion ,

If Grammar is of form

$A \rightarrow A\alpha \mid \beta$

then after removal of left recursion it should be written as

$A \rightarrow \beta A'$

$A' \rightarrow \alpha A' \mid \epsilon$

Since the grammar is :

$E \rightarrow E - T \mid T$ (Here α is '$-T$' and β is T)

$T \rightarrow T + F \mid F$ (Here α is '$+F$' and β is F)

$F \rightarrow (E) \mid id$ (It is not having left recursion)

Rewriting after removing left recursion :

$E \rightarrow TE'$

$E' \rightarrow -TE' \mid \epsilon$

$T \rightarrow FT'$

$T' \rightarrow +FT' \mid \epsilon$

$F \rightarrow (E) \mid id$

Now replace E' with X and T' with Y to match with Option C.

$E \rightarrow TX$

$X \rightarrow -TX \mid \epsilon$

$T \rightarrow FY$

$Y \rightarrow +FY \mid \epsilon$

$F \rightarrow (E) \mid id$

Hence C is correct.

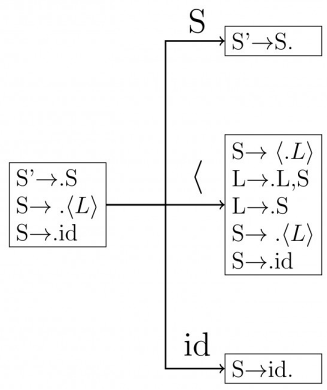
41 votes

-- Indranil Maji (529 points)

2.6.40 Grammar: GATE CSE 2019 | Question: 43 top

https://gateoverflow.in/302805

✔ Total 5 items



👍 23 votes -- Khushboo Mishra (571 points)

2.6.41 Grammar: GATE CSE 2021 Set 1 | Question: 31 top ☛ https://gateoverflow.in/357420

✔

	FIRST	FOLLOW
$S \to daT \mid Rf$	$\{d, c, f\}$	$\{c, f, \$\}$
$T \to aS \mid baT \mid \varepsilon$	$\{a, b, \varepsilon\}$	$\{c, f, \$\}$
$R \to caTR \mid \varepsilon$	$\{c, \varepsilon\}$	$\{f\}$

	a	b	c	d	f	$\$$
S			$S \to Rf$ $\boxed{1}$	$S \to daT$	$S \to Rf$ $\boxed{2}$	
T	$T \to aS$	$T \to baT$	$T \to \varepsilon$ $\boxed{3}$		$T \to \varepsilon$	$T \to \varepsilon$ $\boxed{4}$
R			$R \to caTR$		$R \to \varepsilon$	

Ans: **A** **(1)** $S \to Rf$ **(2)** $S \to Rf$ **(3)** $T \to \varepsilon$ **(4)** $T \to \varepsilon$

👍 3 votes -- Bhargav D Dave (695 points)

2.6.42 Grammar: GATE IT 2007 | Question: 9 top ☛ https://gateoverflow.in/3442

✔ Correct Option: **C**

$L(D) = L(G)$. **Both must represent same language**. Also, if we are converting a grammar from ambiguous to unambiguous form, we must ensure that our new new grammar represents the same language as previous grammar.

For ex $G1 : S \to Sa \mid aS \mid a$; $\{$Ambiguous $(2$ parse trees for string 'aa'$)\}$

$G1' : S \to aS \mid a$; $\{$Unambiguous$\}$

Both represent the language represented by the regular expression: a^+

👍 42 votes -- Anurag Semwal (6.7k points)

2.7 Intermediate Code (10) top

2.7.1 Intermediate Code: GATE CSE 1988 | Question: 2xvii top ☛ https://gateoverflow.in/94350

Construct a DAG for the following set of quadruples:

- E:=A+B
- F:=E-C
- G:=F*D
- H:=A+B
- I:=I-C
- J:=I+G

gate1988 descriptive compiler-design intermediate-code

Answer

2.7.2 Intermediate Code: GATE CSE 1989 | Question: 4-v top https://gateoverflow.in/87885

Is the following code template for the if-then-else statement correct? if not, correct it.

- if expression then statement 1
  else statement 2

Template:

Code for expression

- (*result in $E, E > O$ indicates true *)
- Branch on $E > O$ to $L1$
- Code for statement 1
- $L1$: Code for statement 2

descriptive gate1989 compiler-design intermediate-code

Answer

2.7.3 Intermediate Code: GATE CSE 1992 | Question: 11b top https://gateoverflow.in/43583

Write 3 address intermediate code (quadruples) for the following boolean expression in the sequence as it would be generated by a compiler. Partial evaluation of boolean expressions is not permitted. Assume the usual rules of precedence of the operators.

$$(a+b) > (c+d) \text{ or } a > c \text{ and } b < d$$

gate1992 compiler-design syntax-directed-translation intermediate-code descriptive

Answer

2.7.4 Intermediate Code: GATE CSE 1994 | Question: 1.12 top https://gateoverflow.in/2453

Generation of intermediate code based on an abstract machine model is useful in compilers because

A. it makes implementation of lexical analysis and syntax analysis easier

B. syntax-directed translations can be written for intermediate code generation

C. it enhances the portability of the front end of the compiler

D. it is not possible to generate code for real machines directly from high level language programs

gate1994 compiler-design intermediate-code easy

Answer

2.7.5 Intermediate Code: GATE CSE 2014 Set 2 | Question: 34 top https://gateoverflow.in/1993

For a C program accessing $\mathbf{X[i][j][k]}$, the following intermediate code is generated by a compiler. Assume that the size of an **integer** is 32 bits and the size of a **character** is 8 bits.

```
t0 = i * 1024
t1 = j * 32
t2 = k * 4
t3 = t1 + t0
t4 = t3 + t2
t5 = X[t4]
```

Which one of the following statements about the source code for the C program is CORRECT?

A. $\mathbf{X}$ is declared as "**int** $\mathbf{X[32][32][8]}$".
B. $\mathbf{X}$ is declared as "**int** $\mathbf{X[4][1024][32]}$".
C. $\mathbf{X}$ is declared as "**char** $\mathbf{X[4][32][8]}$".
D. $\mathbf{X}$ is declared as "**char** $\mathbf{X[32][16][2]}$".

gate2014-cse-set2 compiler-design intermediate-code programming-in-c normal

Answer

2.7.6 Intermediate Code: GATE CSE 2014 Set 3 | Question: 17 top

https://gateoverflow.in/2051

One of the purposes of using intermediate code in compilers is to

A. make parsing and semantic analysis simpler.
B. improve error recovery and error reporting.
C. increase the chances of reusing the machine-independent code optimizer in other compilers.
D. improve the register allocation.

gate2014-cse-set3 compiler-design intermediate-code easy

Answer

2.7.7 Intermediate Code: GATE CSE 2015 Set 1 | Question: 55 top

https://gateoverflow.in/8365

The least number of temporary variables required to create a three-address code in static single assignment form for the expression $q + r/3 + s - t * 5 + u * v/w$ is_________________.

gate2015-cse-set1 compiler-design intermediate-code normal numerical-answers

Answer

2.7.8 Intermediate Code: GATE CSE 2015 Set 1 | Question: 8 top

https://gateoverflow.in/8096

For computer based on three-address instruction formats, each address field can be used to specify which of the following:

(S1) A memory operand

(S2) A processor register

(S3) An implied accumulator register

A. Either $S1$ or $S2$
B. Either $S2$ or $S3$
C. Only $S2$ and $S3$
D. All of $S1$, $S2$ and $S3$

gate2015-cse-set1 compiler-design intermediate-code normal

Answer

2.7.9 Intermediate Code: GATE CSE 2015 Set 2 | Question: 29 top

https://gateoverflow.in/8139

Consider the intermediate code given below.

```
(1) i=1
(2) j=1
(3) t1 = 5 * i
(4) t2 = t1 + j
(5) t3 = 4 * t2
(6) t4 = t3
(7) a[t4] = -1
(8) j = j + 1
(9) if j <= 5 goto (3)
(10) i = i +1
(11) if i < 5 goto (2)
```

The number of nodes and edges in control-flow-graph constructed for the above code, respectively, are

A. 5 and 7
B. 6 and 7
C. 5 and 5
D. 7 and 8

gate2015-cse-set2 compiler-design intermediate-code normal

Answer

2.7.10 Intermediate Code: GATE CSE 2021 Set 2 | Question: 13 top

☛ https://gateoverflow.in/357527

In the context of compilers, which of the following is/are NOT an intermediate representation of the source program?

A. Three address code
B. Abstract Syntax Tree (AST)
C. Control Flow Graph (CFG)
D. Symbol table

gate2021-cse-set2 multiple-selects compiler-design intermediate-code

Answer

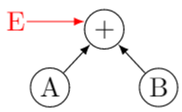
Answers: Intermediate Code

2.7.1 Intermediate Code: GATE CSE 1988 | Question: 2xvii top

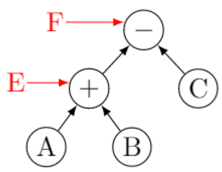
☛ https://gateoverflow.in/94350

✔ The Steps for constructing the DAG are shown below.

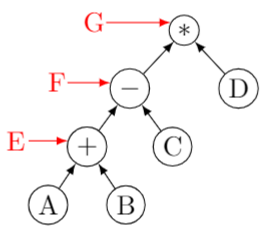
$I.\ E = A + B$



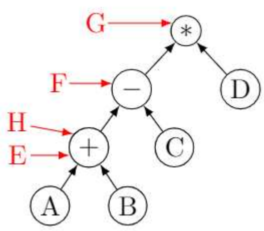
$II.\ F = E - C$



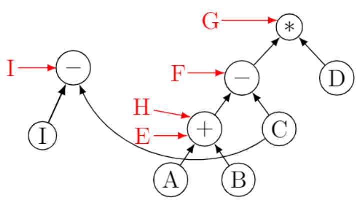
$III.\ G = F * D$



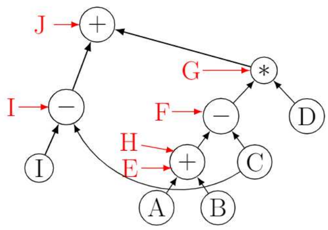
$IV.\ H = A + B$



$V.\ I = I - C$



$VI.\ J = I + G$



👍 16 votes -- Satbir Singh (21k points)

2.7.2 Intermediate Code: GATE CSE 1989 | Question: 4-v top https://gateoverflow.in/87885

✔ The given template is wrong. The following should be correct:

Code for Expression

- Branch on $E > 0$ to L
- Code for segment 2
- Branch to $L1$
- L: Code for statement 1
- $L1$:

👍 12 votes -- Pinaki Dash (1.5k points)

2.7.3 Intermediate Code: GATE CSE 1992 | Question: 11b top https://gateoverflow.in/43583

✔ Each instruction in quadruples presentation is divided into four fields: operator, arg 1, arg2, and result. The above example is represented below in quadruples format:

$(a + b) > (c + d)$ OR $a > c$ AND $b < d$

$(t1 > t2)$ OR $a > c$ AND $b < d$

$t3$ OR $t4$ AND $t5$

$t3$ OR $t6$

$t1 = a + b$

$t2 = c + d$

$t3 = t1 > t2$

$t4 = a > c$

$t5 = b < d$

$t6 = t4$ AND $t5$

$t7 = t3$ OR $t6$

Op	arg1	arg2	Result
+	a	b	t1
+	c	d	t2
>	t1	t2	t3
.>	a	c	t4
<	b	d	t5
AND	t4	t5	t6
OR	t3	t6	t7

👍 38 votes -- shekhar chauhan (32.8k points)

2.7.4 Intermediate Code: GATE CSE 1994 | Question: 1.12 top https://gateoverflow.in/2453

✔ C. stating the actual use of the Intermediate Code.

Also optimizations can be done on intermediate code enhancing the portability of the optimizer.

41 votes -- Gate Keeda (15.9k points)

2.7.5 Intermediate Code: GATE CSE 2014 Set 2 | Question: 34 top https://gateoverflow.in/1993

✔ k is multiplied by 4, means sizeof(dataype) is int.
j is multiplied by 32, means the inner most dimension of array is $32/4 = 8$ (we have to divide by the size of the inner dimension- which here is a simple integer)
i is multiplied by 1024, means the second dimension of array is $1024/32 = 32$ ($32 = 8 * 4$ is the size of the inner dimension here)

So, (**A**) is correct. The first dimension is not needed for code generation and that is why in C language while passing an array to a function, we can omit the value of the first dimension but not any others.

We can also do as follows:

$X[i][j][k] = *(*(*(X + i) + j) + k)$

In Integer arithmetic, this equals

$*(*(*(X + i * sizeof(*X)) + j * sizeof(**X) + k * sizeof(***X))$

as for every add to a pointer we have to multiply the size of the pointed value (to get a valid address)

So, from the given code we get

$sizeof(***X) = 4, -$ int
$sizeof(**X) = 32-$ int array of size 8
$sizeof(*X) = 1024 - 2\ D$ int array of size $[32]$ havinf size of inner $1D$ array 32.

So, the inner dimensions must be 32 and 8 and type must be integer. So, only option A matches.

87 votes -- Arjun Suresh (330k points)

2.7.6 Intermediate Code: GATE CSE 2014 Set 3 | Question: 17 top https://gateoverflow.in/2051

✔ C. that is the actual use of intermediate code generator in a compiler.

32 votes -- Gate Keeda (15.9k points)

2.7.7 Intermediate Code: GATE CSE 2015 Set 1 | Question: 55 top https://gateoverflow.in/8365

✔ Answer is 8.

In compiler design, **static single assignment form** (often abbreviated as **SSA form** or simply **SSA**) is a property of an intermediate representation (IR), which requires that each variable is assigned exactly once, and every variable is defined before it is used. Existing variables in the original IR are split into *versions*, new variables.

We will need a temporary variable for storing the result of each binary operation as SSA (Static Single Assignment) implies the variable cannot be repeated on LHS of assignment.

$q + r/3 + s - t * 5 + u * v/w$

$t1 = r/3;$
$t2 = t * 5;$
$t3 = u * v;$
$t4 = t3/w;$
$t5 = q + t1;$
$t6 = t5 + s;$
$t7 = t6 - t2;$
$t8 = t7 + t4$

http://web.stanford.edu/class/archive/cs/cs143/cs143.1128/handouts/240%20TAC%20Examples.pdf

References

122 votes -- Arjun Suresh (330k points)

2.7.8 Intermediate Code: GATE CSE 2015 Set 1 | Question: 8 top https://gateoverflow.in/8096

✔ **Three address Instruction**

Computer with three addresses instruction format can use each address field to specify either processor register or memory operand.

e.g., $X = (A + B) * (C + A)$

Equivalent **Three address Instructions**

$$\begin{array}{lcl} \text{ADD } R1, A, B & ; & R1 \leftarrow M[A] + M[B] \\ \text{ADD } R2, C, D & ; & R2 \leftarrow M[C] + M[D] \\ \text{MUL } X, R1, R2 & ; & M[X] \leftarrow R1 * R2 \end{array}$$

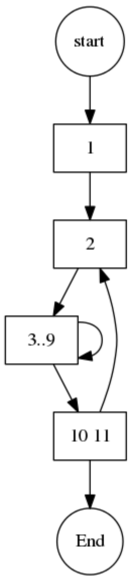
The advantage of the three address formats is that it results in short program when evaluating arithmetic expression. The disadvantage is that the binary-coded instructions require too many bits to specify three addresses.

Correct Answer: A

👍 51 votes -- Prasanna Ranganathan (3.9k points)

2.7.9 Intermediate Code: GATE CSE 2015 Set 2 | Question: 29 top https://gateoverflow.in/8139

✔



Answer is $6, 7$ if we add an explicit start and end nodes. This follows from the definition of CFG in the below IITM link

http://www.cse.iitm.ac.in/~krishna/cs3300/pm-lecture1.pdf

But many of the standard books/universities don't follow this definition.

References

👍 49 votes -- Arjun Suresh (330k points)

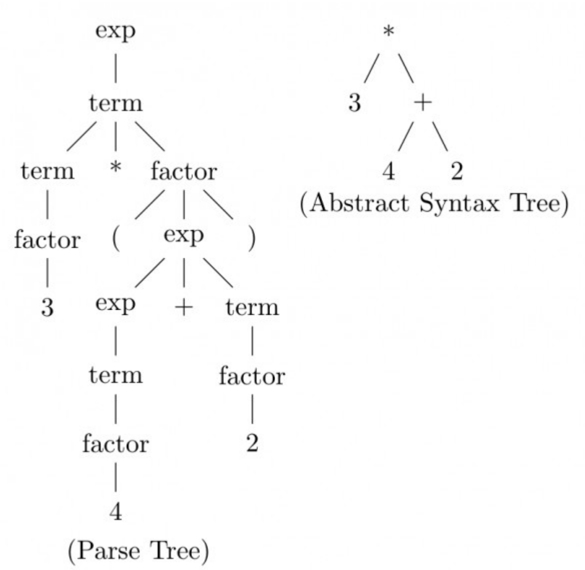
2.7.10 Intermediate Code: GATE CSE 2021 Set 2 | Question: 13 top https://gateoverflow.in/357527

✔ **Correct Answer: D**

Symbol table is a data structure created and maintained by compilers in order to store info about occurrences of various entities like variable names, function names, objects, classes and interface.

Various forms of intermediate representation of code include **Postfix Notation**, **Three address code** (**x** = **y** op **z**), **Syntax Tree**, **DAG.**
Abstract Syntax Tree is a condensed version of **syntax tree/parse tree** more to with program than the compiler.

Parse Tree and Syntax Tree:



(Parse Tree) (Abstract Syntax Tree)

Control Flow Graph is used in optimization phase of compiler,each basic block consists of **linear code**, the next block to access is determined by the **last instruction** of the current block.

An Example,

```
        goto L2
L1:
        t0 := 3 >> x
        t1 := y / t0
        x := t1
        if y == 0 goto L3
        t2 := x - 3
        print t2
L3:
L2:
        t4 := 4 * y
        x := t4 < t5
        if t5 != 0 goto L1
```

See:

1. https://cs.lmu.edu/~ray/notes/ir/
2. https://www2.cs.arizona.edu/~collberg/Teaching/453/2009/Handouts/Handout-15.pdf
3. http://pages.cs.wisc.edu/~fischer/cs536.s06/course.hold/html/NOTES/4.SYNTAX-DIRECTED-TRANSLATION.html

References

5 votes -- Cringe is my middle name... (817 points)

2.8 Lexical Analysis (6) top

2.8.1 Lexical Analysis: GATE CSE 1987 | Question: 1-xvii top https://gateoverflow.in/80364

Using longer identifiers in a program will necessarily lead to:

A. Somewhat slower compilation
B. A program that is easier to understand
C. An incorrect program
D. None of the above

gate1987 compiler-design lexical-analysis

Answer

2.8.2 Lexical Analysis: GATE CSE 2000 | Question: 1.18, ISRO2015-25 top https://gateoverflow.in/641

The number of tokens in the following C statement is

```
printf("i=%d, &i=%x", i, &i);
```

A. 3
B. 26
C. 10
D. 21

gate2000-cse compiler-design lexical-analysis easy isro2015

Answer

2.8.3 Lexical Analysis: GATE CSE 2010 | Question: 13 top https://gateoverflow.in/2186

Which data structure in a compiler is used for managing information about variables and their attributes?

A. Abstract syntax tree
B. Symbol table
C. Semantic stack
D. Parse table

gate2010-cse compiler-design lexical-analysis easy

Answer

2.8.4 Lexical Analysis: GATE CSE 2011 | Question: 1 top https://gateoverflow.in/2103

In a compiler, keywords of a language are recognized during

A. parsing of the program
B. the code generation
C. the lexical analysis of the program
D. dataflow analysis

gate2011-cse compiler-design lexical-analysis easy

Answer

2.8.5 Lexical Analysis: GATE CSE 2011 | Question: 19 top https://gateoverflow.in/2121

The lexical analysis for a modern computer language such as Java needs the power of which one of the following machine models in a necessary and sufficient sense?

A. Finite state automata
B. Deterministic pushdown automata
C. Non-deterministic pushdown automata
D. Turing machine

gate2011-cse compiler-design lexical-analysis easy

Answer

2.8.6 Lexical Analysis: GATE CSE 2018 | Question: 37 top https://gateoverflow.in/204111

A lexical analyzer uses the following patterns to recognize three tokens T_1, T_2, and T_3 over the alphabet $\{a, b, c\}$.

- T_1: $a?(b \mid c)^*a$
- T_2: $b?(a \mid c)^*b$
- T_3: $c?(b \mid a)^*c$

Note that 'x?' means 0 or 1 occurrence of the symbol x. Note also that the analyzer outputs the token that matches the longest possible prefix.

If the string $bbaacabc$ is processed by the analyzer, which one of the following is the sequence of tokens it outputs?

A. $T_1T_2T_3$

B. $T_1T_1T_3$
C. $T_2T_1T_3$
D. T_3T_3

gate2018-cse compiler-design lexical-analysis normal

Answer

Answers: Lexical Analysis

2.8.1 Lexical Analysis: GATE CSE 1987 | Question: 1-xvii top

https://gateoverflow.in/80364

✔ **Answer**: **Option A**) is Correct because lex will take more time to recognize the longer identifiers.

15 votes -- sarveswara rao vangala (1.4k points)

2.8.2 Lexical Analysis: GATE CSE 2000 | Question: 1.18, ISRO2015-25 top

https://gateoverflow.in/641

✔ answer - **C**

Tokens are:

1. `printf`
2. `(`
3. `"i=%d, &i=%x"`
4. `,`
5. `i`
6. `,`
7. `&`
8. `i`
9. `)`
10. `;`

47 votes -- Ankit Rokde (6.9k points)

2.8.3 Lexical Analysis: GATE CSE 2010 | Question: 13 top

https://gateoverflow.in/2186

✔ (B) **Symbol table** is the answer.

It can be implemented by using an array, hash table, tree and even some time with the help of the Iinked list!

30 votes -- Dexter (7.1k points)

2.8.4 Lexical Analysis: GATE CSE 2011 | Question: 1 top

https://gateoverflow.in/2103

✔ Typically, the lexical analysis phase of compilation breaks the input text up into sequences of lexemes that each belongs to some particular token type that's useful in later analysis. Consequently, keywords are usually first recognized during lexical analysis in order to make parsing easier. Since parsers tend to be implemented by writing context-free grammars of tokens rather than of lexemes (that is, the *category* of the lexeme rather than the *contents* of the lexeme), it is significantly easier to build a parser when keywords are marked during lexing. Any identifier is also a token so it is recognized in lexical Analysis .
Hence, option **C** is True.
ref@ http://stackoverflow.com/questions/5202709/phases-of-a-compiler

References

45 votes -- Mithlesh Upadhyay (4.3k points)

2.8.5 Lexical Analysis: GATE CSE 2011 | Question: 19 top

https://gateoverflow.in/2121

✔ Answer - **A**

In compiler lexical analyzer categorizes character sequence into lexemes and produces tokens as output for parser. And tokens are expressed in regular expressions so a simple Finite Automata is sufficient for it.

41 votes -- Ankit Rokde (6.9k points)

2.8.6 Lexical Analysis: GATE CSE 2018 | Question: 37 top https://gateoverflow.in/204111

✔ **Option D** is the correct answer.

You can think T_3 as $(\varepsilon + c)\,(b+a)^{*}c$

Given string is $bbaacabc$

The longest matching prefix bbaac { from regex $T3$ you can easily derive bbaac }

Now the remaining abc { This can also be derived from $T3$ }

Hence $T3T3$ is the answer.

38 votes -- Ruturaj Mohanty (3.1k points)

2.9 Linker (3) top

2.9.1 Linker: GATE CSE 1991 | Question: 03,ix top https://gateoverflow.in/519

A "link editor" is a program that:

A. matches the parameters of the macro-definition with locations of the parameters of the macro call
B. matches external names of one program with their location in other programs
C. matches the parameters of subroutine definition with the location of parameters of subroutine call.
D. acts as a link between text editor and the user
E. acts as a link between compiler and the user program

gate1991 compiler-design normal linker multiple-selects

Answer

2.9.2 Linker: GATE CSE 2003 | Question: 76 top https://gateoverflow.in/962

Which of the following is NOT an advantage of using shared, dynamically linked libraries as opposed to using statistically linked libraries?

A. Smaller sizes of executable files

B. Lesser overall page fault rate in the system

C. Faster program startup

D. Existing programs need not be re-linked to take advantage of newer versions of libraries

gate2003-cse compiler-design runtime-environments linker easy

Answer

2.9.3 Linker: GATE CSE 2004 | Question: 9 top https://gateoverflow.in/1006

Consider a program P that consists of two source modules M_1 and M_2 contained in two different files. If M_1 contains a reference to a function defined in M_2 the reference will be resolved at

A. Edit time
B. Compile time
C. Link time
D. Load time

gate2004-cse compiler-design easy linker

Answer

Answers: Linker

2.9.1 Linker: GATE CSE 1991 | Question: 03,ix top https://gateoverflow.in/519

✔ Link editor or (linker) performs

1. external symbol resolution
2. relocation.

ANS: **B**

Matches external names of one program with their location in other programs.

35 votes -- pramod (2.8k points)

2.9.2 Linker: GATE CSE 2003 | Question: 76 top https://gateoverflow.in/962

✔ **option C:** DLL takes more time in program setup (in loading and linking phase to set up the global offset table and load and link the required libraries)

- Since DLLs are separated from executable, the size of executable becomes smaller. Since DLLs are shared among multiple executables, the total memory usage of the system goes down and hence overall page fault rate decreases.
- Dynamic linking takes place during program runtime. So, if a DLL is replaced to a new version, it will automatically get linked during runtime. There is no explicit relinking required as in the case of static linking. (This works by linking the DLL calls to Global Offset Table and the contents of this table is filled during program run. A simple jump in static linking becomes an indirect jump in dynamic linking).

Refer: Galvin section 8.1.5, Dynamic Linking and Shared Libraries

67 votes -- GateMaster Prime (1.2k points)

2.9.3 Linker: GATE CSE 2004 | Question: 9 top https://gateoverflow.in/1006

✔ answer - **C**. Each module is compiled separately and then linked together to make the executable. The below commands shows how to do this for two modules $c1.c$ and $c2.c$ using gcc.

```
gcc -c c1.c -o c1.o
gcc -c c2.c -o c2.o
gcc c1.o c2.o -o C.exe
```

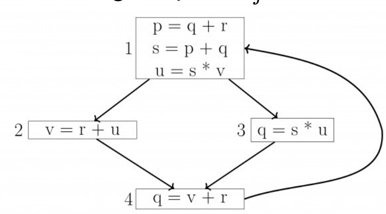
47 votes -- Ankit Rokde (6.9k points)

2.10 Live Variable (1) top

2.10.1 Live Variable: GATE CSE 2015 Set 1 | Question: 50 top https://gateoverflow.in/8356

A variable x is said to be live at a statement s_i in a program if the following three conditions hold simultaneously:

i. There exists a statement S_j that uses x
ii. There is a path from S_i to S_j in the flow graph corresponding to the program
iii. The path has no intervening assignment to x including at S_i and S_j



The variables which are live both at the statement in basic block 2 and at the statement in basic block 3 of the above control flow graph are

A. p, s, u
B. r, s, u
C. r, u
D. q, v

gate2015-cse-set1 compiler-design live-variable normal

Answer

Answers: Live Variable

2.10.1 Live Variable: GATE CSE 2015 Set 1 | Question: 50 top

☛ https://gateoverflow.in/8356

✔ **r, u.**

p, and s are assigned to in 1 and there is no intermediate use of them before that. Hence p, and s are not live in both 2 and 3.
q is assigned to in 4 and hence is not live in both 2 and 3.

v is live at 3 but not at 2.

u is live at 3 and also at 2 if we consider a path of length 0 from $2-2$.

So, r, u is the answer.

👍 33 votes -- Arjun Suresh (330k points)

2.11 Lr Parser (1) top

2.11.1 Lr Parser: GATE CSE 2021 Set 1 | Question: 5 top

☛ https://gateoverflow.in/357447

Consider the following statements.

- S_1 : Every $\text{SLR}(1)$ grammar is unambiguous but there are certain unambiguous grammars that are not $\text{SLR}(1)$.
- S_2 : For any context-free grammar, there is a parser that takes at most $O(n^3)$ time to parse a string of length n.

Which one of the following options is correct?

A. S_1 is true and S_2 is false
B. S_1 is false and S_2 is true
C. S_1 is true and S_2 is true
D. S_1 is false and S_2 is false

gate2021-cse-set1 compiler-design lr-parser

Answer

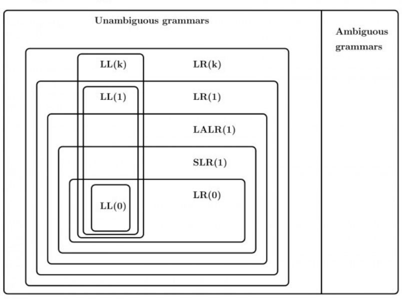
Answers: Lr Parser

2.11.1 Lr Parser: GATE CSE 2021 Set 1 | Question: 5 top

☛ https://gateoverflow.in/357447

✔ Correct option is C. Both statements are correct.
An unambiguous grammar is not necessarily $\text{SLR}(1)$. But every $\text{SLR}(1)$ grammar is unambiguous.

LL(1) versus LR(k)



We do have
CYK algorithm which takes $O(n^3)$ time (assuming size of the context-free grammar $|G|$ to be a constant) to parse any string of length n using a context-free grammar G.

References

👍 7 votes -- Ankur tiwari (557 points)

2.12 Lr Parsing (2) top

2.12.1 Lr Parsing: GATE CSE 2020 | Question: 24 top https://gateoverflow.in/333207

Consider the following grammar.

- $S \to aSB \mid d$
- $B \to b$

The number of reduction steps taken by a bottom-up parser while accepting the string $aaadbbb$ is ________.

gate2020-cse numerical-answers compiler-design lr-parsing

Answer

2.12.2 Lr Parsing: GATE CSE 2021 Set 2 | Question: 51 top https://gateoverflow.in/357486

Consider the following augmented grammar with $\{\#, @, <, >, a, b, c\}$ as the set of terminals.

$$
\begin{gathered}
S' \to S \\
S \to S\#cS \\
S \to SS \\
S \to S \quad @ \\
S \to < S > \\
S \to a \\
S \to b \\
S \to c
\end{gathered}
$$

Let $I_0 = \text{CLOSURE}(\{S' \to \bullet S\})$. The number of items in the set $\text{GOTO}(\text{GOTO}(I_0 <), <)$ is ________

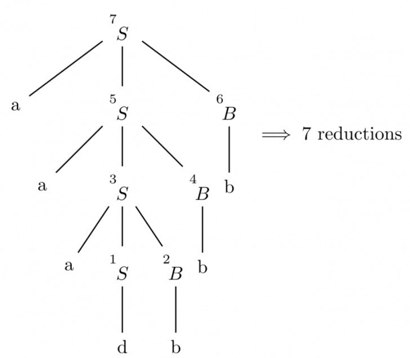
gate2021-cse-set2 compiler-design lr-parsing numerical-answers

Answer

Answers: Lr Parsing

2.12.1 Lr Parsing: GATE CSE 2020 | Question: 24 top https://gateoverflow.in/333207

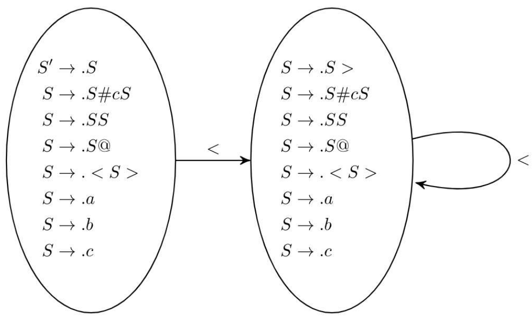
✔ In parse tree, all the non terminals are reductions. So total 7 reductions.



👍 12 votes -- Navneet Singh Tomar (789 points)

2.12.2 Lr Parsing: GATE CSE 2021 Set 2 | Question: 51 top https://gateoverflow.in/357486

✔



$I_0 = \text{Closure}(S' \to .S)$

We can count the items in the third collection.

Answer : 8

👍 3 votes -- shashankpal (523 points)

2.13 Macros (4) top

2.13.1 Macros: GATE CSE 1992 | Question: 01,vii top https://gateoverflow.in/552

Macro expansion is done in pass one instead of pass two in a two pass macro assembler because _________

gate1992 compiler-design macros easy fill-in-the-blanks

Answer

2.13.2 Macros: GATE CSE 1995 | Question: 1.11 top https://gateoverflow.in/2598

What are x and y in the following macro definition?

```
macro Add x, y
        Load y
        Mul x
        Store y
end macro
```

A. Variables
B. Identifiers
C. Actual parameters
D. Formal parameters

gate1995 compiler-design macros easy

Answer

2.13.3 Macros: GATE CSE 1996 | Question: 2.16 top https://gateoverflow.in/2745

Which of the following macros can put a macro assembler into an infinite loop?

i.
```
.MACRO M1, X
.IF EQ, X   ;if X=0 then
M1 X + 1
.ENDC
.IF NE, X   ;if X ≠ O then
.WORD X  ;address (X) is stored here
.ENDC
.ENDM
```

ii.
```
.MACRO M2, X
.IF EQ, X
M2 X
.ENDC
.IF NE, X
.WORD X + 1
.ENDC
.ENDM
```

A. (ii) only
B. (i) only

C. both (i) and (ii)
D. None of the above

gate1996 compiler-design macros normal

Answer

2.13.4 Macros: GATE CSE 1997 | Question: 1.9 top

https://gateoverflow.in/2225

The conditional expansion facility of macro processor is provided to

A. test a condition during the execution of the expanded program
B. to expand certain model statements depending upon the value of a condition during the execution of the expanded program
C. to implement recursion
D. to expand certain model statements depending upon the value of a condition during the process of macro expansion

gate1997 compiler-design macros easy

Answer

Answers: Macros

2.13.1 Macros: GATE CSE 1992 | Question: 01,vii top

https://gateoverflow.in/552

in two processor micro processor all the macro definitions are processed during the first pass it self. so macro expansions are done in pass 1 only.

0 votes -- manoranjan gr (11 points)

2.13.2 Macros: GATE CSE 1995 | Question: 1.11 top

https://gateoverflow.in/2598

ans is **D**

- **formal parameter** — the identifier used in a method to stand for the value that is passed into the method by a caller.
  - For example, `amount` is a formal parameter of `calculate`
- **actual parameter** — the actual value that is passed into the method by a caller.
  - For example, the `800` used when `calculate` is called is an actual parameter.
  - actual parameters are often called **arguments**

```
float calculate (float amount)
{
 return amount * 1.2;
}

int main()
{
  ...
  float final = calculate(800);
  ...
}
```

14 votes -- jayendra (6.7k points)

2.13.3 Macros: GATE CSE 1996 | Question: 2.16 top

https://gateoverflow.in/2745

✔ If $M2$ macro is called with $X = 0$, then the macro assembler will go into an infinite loop.

For $M1$ the argument is incremented for the recursive call and so the macro expansion will happen maximum 2 times.

Hence, correct option: **A**.

10 votes -- suraj (4.8k points)

2.13.4 Macros: GATE CSE 1997 | Question: 1.9 top

https://gateoverflow.in/2225

✔ Macro is expanded during the process of macro expansion. Hence, correct answer would be (d).

17 votes -- suraj (4.8k points)

2.14 Parameter Passing (14) top

2.14.1 Parameter Passing: GATE CSE 1988 | Question: 2xv top

☛ https://gateoverflow.in/94333

What is printed by following program, assuming call-by reference method of passing parameters for all variables in the parameter list of procedure P?

```
program     Main(inout, output);
var     a, b:integer;
        procedure P(x, y, z:integer);
        begin
            y:=y+1
            z:=x+x
        end P;
begin
    a:=2; b:=3;
    p(a+b, a, a);
    Write(a)
end.
```

gate1988 descriptive compiler-design runtime-environments parameter-passing numerical-answers

Answer

2.14.2 Parameter Passing: GATE CSE 1988 | Question: 8i top

☛ https://gateoverflow.in/94371

Consider the procedure declaration:

```
Procedure
P (k: integer)
```

where the parameter passing mechanism is call-by-value-result. Is it correct if the call, P (A[i]), where A is an array and i an integer, is implemented as below.

a. create a new local variable, say z;
b. assign to z, the value of A [i];
c. execute the body of P using z for k;
d. set A [i] to z;

Explain your answer. If this is incorrect implementation, suggest a correct one.

gate1988 descriptive compiler-design runtime-environments parameter-passing

Answer

2.14.3 Parameter Passing: GATE CSE 1989 | Question: 3-viii top

☛ https://gateoverflow.in/37264

In which of the following case(s) is it possible to obtain different results for call-by-reference and call-by-name parameter passing?

A. Passing an expression as a parameter
B. Passing an array as a parameter
C. Passing a pointer as a parameter
D. Passing as array element as a parameter

gate1989 parameter-passing runtime-environments compiler-design multiple-selects

Answer

2.14.4 Parameter Passing: GATE CSE 1990 | Question: 11a top

☛ https://gateoverflow.in/85981

What does the following program output?

```
program module (input, output);
var
    a:array [1...5] of integer;
    i, j: integer;
procedure unknown (var b: integer, var c: integer);
var
    i:integer;
begin
    for i := 1 to 5 do a[i] := i;
    b:= 0; c := 0
    for i := 1 to 5 do write (a[i]);
```

```
    writeln();
    a[3]:=11; a[1]:=11;
    for i:=1 to 5 do a [i] := sqr(i);
    writeln(c,b); b := 5; c := 6;
end;
begin
  i:=1; j:=3; unknown (a[i], a[j]);
  for i:=1 to 5 do write (a[i]);
end;
```

gate1990 descriptive compiler-design runtime-environments parameter-passing

Answer

2.14.5 Parameter Passing: GATE CSE 1991 | Question: 03,x top

https://gateoverflow.in/524

Indicate all the true statements from the following:

A. Recursive descent parsing cannot be used for grammar with left recursion.
B. The intermediate form for representing expressions which is best suited for code optimization is the postfix form.
C. A programming language not supporting either recursion or pointer type does not need the support of dynamic memory allocation.
D. Although C does not support call-by-name parameter passing, the effect can be correctly simulated in C
E. No feature of Pascal typing violates strong typing in Pascal.

gate1991 compiler-design parameter-passing difficult multiple-selects

Answer

2.14.6 Parameter Passing: GATE CSE 1991 | Question: 09a top

https://gateoverflow.in/536

Consider the following pseudo-code (all data items are of type integer):

```
procedure P(a, b, c);
    a := 2;
    c := a + b;
end {P}

begin
    x := 1;
    y := 5;
    z := 100;
    P(x, x*y, z);
    Write ('x = ', x, 'z = ', z);
end
```

Determine its output, if the parameters are passed to the Procedure P by

i. value
ii. reference
iii. name

gate1991 compiler-design parameter-passing normal runtime-environments descriptive

Answer

2.14.7 Parameter Passing: GATE CSE 1991 | Question: 09b top

https://gateoverflow.in/43603

For the following code, indicate the output if

a. static scope rules
b. dynamic scope rules

are used

```
var a,b : integer;

procedure P;
    a := 5;
    b := 10;
end {P};

procedure Q;
```

```
    var a, b : integer;
    P;
end {Q};

begin
    a := 1;
    b := 2;
    Q;
    Write ('a = ', a, 'b = ', b);
end
```

gate1991 runtime-environments normal compiler-design parameter-passing descriptive

Answer

2.14.8 Parameter Passing: GATE CSE 1993 | Question: 26 top

https://gateoverflow.in/2322

A stack is used to pass parameters to procedures in a procedure call.

A. If a procedure P has two parameters as described in procedure definition:

```
procedure P (var x :integer; y: integer);
```

and if P is called by ; $P(a, b)$

State precisely in a sentence what is pushed on stack for parameters a and b

B. In the generated code for the body of procedure P, how will the addressing of formal parameters x and y differ?

gate1993 compiler-design parameter-passing runtime-environments normal descriptive

Answer

2.14.9 Parameter Passing: GATE CSE 1995 | Question: 2.4 top

https://gateoverflow.in/2616

What is the value of X printed by the following program?

```
program COMPUTE (input, output);
var X:integer;
procedure FIND (X:real);
    begin
        X:=sqrt(X);
    end;
begin
    X:=2
    FIND(X);
    writeln(X);
end.
```

A. 2
B. $\sqrt{2}$
C. Run time error
D. None of the above

gate1995 compiler-design parameter-passing runtime-environments easy

Answer

2.14.10 Parameter Passing: GATE CSE 1999 | Question: 15 top

https://gateoverflow.in/1514

What will be the output of the following program assuming that parameter passing is

i. call by value
ii. call by reference
iii. call by copy restore

```
procedure P{x, y, z};
begin
    y:y+1;
    z: x+x;
end;
begin
    a:= b:= 3;
    P(a+b, a, a);
```

```
        Print(a);
    end
```

gate1999 parameter-passing normal runtime-environments descriptive

Answer

2.14.11 Parameter Passing: GATE CSE 2003 | Question: 74 top

https://gateoverflow.in/43575

The following program fragment is written in a programming language that allows global variables and does not allow nested declarations of functions.

```
global int i=100, j=5;
void P(x) {
 int i=10;
    print(x+10);
    i=200;
    j=20;
    print (x);
}
main() {P(i+j);}
```

If the programming language uses dynamic scoping and call by name parameter passing mechanism, the values printed by the above program are

A. $115, 220$
B. $25, 220$
C. $25, 15$
D. $115, 105$

gate2003-cse programming compiler-design parameter-passing runtime-environments normal

Answer

2.14.12 Parameter Passing: GATE CSE 2004 | Question: 2,ISRO2017-54 top

https://gateoverflow.in/999

Consider the following function

```
void swap(int a, int b)
{
    int temp;
    temp = a;
    a = b;
    b = temp;
}
```

In order to exchange the values of two variables x and y.

A. call $swap(x, y)$
B. call $swap(\&x, \&y)$
C. $swap(x, y)$ cannot be used as it does not return any value
D. $swap(x, y)$ cannot be used as the parameters are passed by value

gate2004-cse compiler-design programming-in-c parameter-passing easy isro2017 runtime-environments

Answer

2.14.13 Parameter Passing: GATE CSE 2016 Set 1 | Question: 36 top

https://gateoverflow.in/39701

What will be the output of the following pseudo-code when parameters are passed by reference and dynamic scoping is assumed?

```
a = 3;
void n(x) { x = x * a; print (x); }
void m(y) { a = 1 ; a = y - a;  n(a); print (a); }
void main () { m(a); }
```

A. $6, 2$
B. $6, 6$
C. $4, 2$
D. $4, 4$

gate2016-cse-set1 parameter-passing normal

Answer

2.14.14 Parameter Passing: GATE IT 2007 | Question: 33 top

https://gateoverflow.in/3466

Consider the program below in a hypothetical language which allows global variable and a choice of call by reference or call by value methods of parameter passing.

```
int i ;
program main ()
{
    int j = 60;
    i = 50;
    call f (i, j);
    print i, j;
}
procedure f (x, y)
{
    i = 100;
    x = 10;
    y = y + i ;
}
```

Which one of the following options represents the correct output of the program for the two parameter passing mechanisms?

A. Call by value : $i = 70, j = 10$; Call by reference :$i = 60, j = 70$
B. Call by value : $i = 50, j = 60$; Call by reference :$i = 50, j = 70$
C. Call by value : $i = 10, j = 70$; Call by reference :$i = 100, j = 60$
D. Call by value : $i = 100, j = 60$; Call by reference :$i = 10, j = 70$

gate2007-it programming parameter-passing normal compiler-design runtime-environments

Answer

Answers: Parameter Passing

2.14.1 Parameter Passing: GATE CSE 1988 | Question: 2xv top

https://gateoverflow.in/94333

✔ let variable "a" has address 100 and "b" has 200 .

and a variable in which "$a + b$" is stored has address 300.

now $p(300, 100, 100)$ which represent x, y, z

$y := y + 1$ // it makes $a = 3$;

$z := x + x$ // x means the value contained at address 300 i.e. 5

$5 + 5 = 10$ hence value at address 100 i.e. variable "a" will get the value 10 .

Hence the value of a i.e. 10 will be printed.

19 votes -- psb (617 points)

2.14.2 Parameter Passing: GATE CSE 1988 | Question: 8i top

https://gateoverflow.in/94371

The given implementation is correct.

In call-by-value-result parameter passing mechanism, formal parameter gets its initial value from the actual parameter being passed. During execution of the function, there is no effect on the actual argument but once the function is returned, the final value of the parameter is copied back to the actual parameter. And this is exactly what the given implementation does.

This is different from call-by-reference in the sense that there whatever modification happens to the parameter in a function, it must reflect in the actual argument too. These two can produce different outputs if a global variable is being used as a parameter.

4 votes -- Arjun Suresh (330k points)

2.14.3 Parameter Passing: GATE CSE 1989 | Question: 3-viii top https://gateoverflow.in/37264

✔ Answer A, D.

A is correct as call-by-name works like a macro and substitution happens only during use time. For example if we pass $2+3$ to the below function

```
int foo(int x)
{
    return x * x;
}
```

we get $2+3*2+3$ which will be 11 due to the higher precedence for $*$. But, call by reference will return $5*5=25$. (For call by reference, when an expression is passed, a temporary variable is created and passed to the function)

D is also correct: Passing an array element as a parameter

See the below example:

```
void m(int x,int y){
    for(int k = 0;k < 10;k++){
        y = 0; x++;
    }
}

int main(){
    int j; int A[10];
    j = 0;
    m(j,A[j]);
    return 0;
}
```

For the above example if we use 'Call by name' its initialize all the array elements with 0. But if we apply ' Call by Reference ' it will only initialize $A[0]$ with 0.

👍 18 votes -- mystylecse (1.8k points)

2.14.4 Parameter Passing: GATE CSE 1990 | Question: 11a top https://gateoverflow.in/85981

✔ We can assume that the code is using pass-by-reference or else there is no use in the parameters b, c.

```
program module (input, output);
var
    a:array [1...5] of integer;
    i, j: integer;
procedure unknown (var b: integer, var c: integer);
//pass by reference; b and a[1] have same address
//and c and a[3] have same address
var
    i:integer;
begin
    for i := 1 to 5 do a[i] := i;
    b:= 0; c := 0
    for i := 1 to 5 do write (a[i]); // prints 0, 2, 0, 4, 5
    writeln();
    a[3]:=11; a[1]:=11;
    for i:=1 to 5 do a [i] := sqr(i);
    writeln(c,b); // prints 121, 121
    b := 5; c := 6;
end;
begin
  i:=1; j:=3; unknown (a[i], a[j]); //pass by reference
  for i:=1 to 5 do write (a[i]); //prints 5, 4, 6, 16, 25
end;
```

👍 1 votes -- Arjun Suresh (330k points)

2.14.5 Parameter Passing: GATE CSE 1991 | Question: 03,x top https://gateoverflow.in/524

✔

A. is TRUE. Left recursive grammars if used directly in recursive descent parsing causes an infinite loop. So, left recursion must be removed before giving to a recursive descent parser.

B. is a strong statement- but I do not have any proof or reference for this- so for now I consider this FALSE.

C. is false. The language can have dynamic data types which requires dynamically growing memory when data type size increases.

D. is true and using macro we can do this.

E. out of syllabus now.

25 votes -- Arjun Suresh (330k points)

2.14.6 Parameter Passing: GATE CSE 1991 | Question: 09a top https://gateoverflow.in/536

✔

1. Pass by value: Function cannot modify a variable in the calling function. So,
$x = 1, z = 100$

2. Pass by reference: An alias of the variable (a different name but having the same memory location) is used to pass the variable to a function. So, whatever change occurs for the variable in the called function is reflected in the calling function.
$x = 2, z = 7(2+5)$

3. Pass by name: The expression used to call a function is copy-pasted for each formal parameter. So, the body of P becomes,
$x := 2;$
$z := x + x * y;$

So, printed value will be b$x = 2, z = 12$

36 votes -- Arjun Suresh (330k points)

2.14.7 Parameter Passing: GATE CSE 1991 | Question: 09b top https://gateoverflow.in/43603

✔ In static scoping, if a variable is not found (variable definition - memory allocation) in the local scope (current function, which includes the current block, then parent block etc.), it is looked upon in global scope. In dynamic scoping, if a variable is not found in local scope, it is looked upon in the function which called the current executing one.

1. $a = 5, b = 10$. main is using global variables. P is also using global variables.
2. $a = 1, b = 2$. main is using global variables. P is using the local variables defined in Q.

(The modification in Q, happens to the variables in P but in main we use the global variables)

22 votes -- Arjun Suresh (330k points)

2.14.8 Parameter Passing: GATE CSE 1993 | Question: 26 top https://gateoverflow.in/2322

a is pointer variable so address and b is variable so its value pushed into stack.

5 votes -- Digvijay (44.9k points)

2.14.9 Parameter Passing: GATE CSE 1995 | Question: 2.4 top https://gateoverflow.in/2616

Answer should be **A**.

As per call by value concept. X in the procedure FIND is a local variable and so no change will be reflected in the global var X.

15 votes -- jayendra (6.7k points)

2.14.10 Parameter Passing: GATE CSE 1999 | Question: 15 top https://gateoverflow.in/1514

✔

- Call by Value : 3
- Call by Reference : 12
- Call by Copy-Restore : 12

The 3 parameter passing mechanisms are simulated in the following 'C' codes.

PS: C language only supports Call by Value and even in the case of pointers, the value of the pointer is getting passed explicitly in a pointer variable. This is different from call by reference (say in C++) where this happens implicitly. The following code for Call-by-Reference and Call-by-copy-restore is just a simulation of the parameter passing behaviour and their implementation in any language need not be the same.

1. Call by Value

```
#include <stdio.h>
int foo(int x,int y,int z)
{
    y = y+1;
    z = x+x;
}
int main(void)
{
    int a = 3;
    int b = 3;
    foo(a+b,a,a);
    printf("%d",a);
    return 0;
}
```

2. Call by Reference (Call by reference is simulated by passing address in C) :

```
#include <stdio.h>
int foo(int *x,int *y,int *z)
{
    *y = *y+1;
    *z = *x+*x;
}
int main(void)
{
    int a = 3;
    int b = 3;
    int c = a+b;
    foo(&c,&a,&a);
    printf("%d",a);
    return 0;
}
```

3. Call by Copy-Restore:

```
#include <stdio.h>
void foo(int *x,int *y,int *z)
{
    *y = *y+1;
    *z = *x+*x;
}
int main(void)
{
    int a=3;
    int b=3;
    int c=a+b;
    int d,e;
    d = c;//copy
    e = a;//copy
    foo(&d,&e,&e);
    a = e;//restore
    c = d;//restore
    printf("%d",a);
    return 0;
}
```

17 votes -- Aditya Gaurav (2.4k points)

2.14.11 Parameter Passing: GATE CSE 2003 | Question: 74 top

https://gateoverflow.in/43575

✔ Answer : No Option is correct.

Answer to this question can be found in the Example 6 below.

"Call by name" parameter passing technique was used by some imperative languages like Algol W, and it is used by several functional languages, like Haskell.

We'll see how "Call by name" parameter passing technique works(theoretical Idea for understanding its working), Not how it is actually implemented(practical implementation).

How "Call by name" works (Idea of this technique) :

In general, the effect of pass-by-name is to **textually substitute** the argument expressions(actual parameters) in a procedure call for the corresponding parameters(formal parameters) in the body of the called procedure.*

i.e. Direct Substitution of actual parameters in the place of formal parameter in the called procedure.*

(* means that it is not complete definition/statement and some technical details are missing and as we go on, we'll fill in these details.)

Example 1 : What will be the output if "Call by name" parameter passing technique is used.

Given Program :

```
void P(x) {
    print(x+10);
    print (x);
}
main()
{
    int j = 10;
    P(j);
}
```

Answer : Since "Call by name" parameter passing technique is used, we can re-write the program as following :

```
void P(x) {
 print(j+10);
    print (j);
}
main()
{
int j = 10;
P(j);
}
```

Hence, output : 20,10

> NOTE 1 (Technical Detail 1) : It does Not matter which Scoping is used(static or dynamic),
> once we substitute actual arguments in the place of formal parameters in the called function,
> for those variables in the actual arguments, environment of caller function will be applicable.

Hence, in the above example 1, in function P, j refers to the j of the caller function i.e. main function.

Example 2 will illustrate the Note 1.

Example 2 : What will be the output if "Call by name" parameter passing technique is used in case of static and dynamic scopings?

```
global int j=100;
void P(x) {
 print(x+10);
    print (x);
}
main()
{
int j = 10;
P(j);
}
```

Answer : Since "Call by name" parameter passing technique is used, we can re-write the program as following :

```
global int j=100;
void P(x) {
 print(j+10);
    print (j);
}
main()
{
int j = 10;
P(j);
}
```

In case of Dynamic scoping : 20,10

In case of Static scoping : 20,10

Note that in case of Static scoping, j in function P does not refer to the global variable But j refers to the caller function, more precisely, Once we substitute Actual arguments in the place of formal parameters in the called function, for those variables in the actual arguments, **Environment of caller function will be applicable.** Hence, j in P will be accessed/updated accroding to caller function's environment and since, caller function i.e. main function here, has a local variable j, so, this j will be accessed/updated by function P.

Example 3 : What will be the output if "Call by name" parameter passing technique is used in case of static and dynamic scopings?

```
global int j=100;
void P(x) {
 print(x+10);
    print (x);
}
main()
{
P(j);
}
```

Answer : Since "Call by name" parameter passing technique is used, we can re-write the program as following :

```
global int j=100;
void P(x) {
    print(j+10);
    print (j);
}
main()
{
    P(j);
}
```

In case of Dynamic scoping : 110,100

In case of Static scoping : 110,100

Again, Once we substitute Actual arguments in the place of formal parameters in the called function, for those variables in the actual arguments, **Environment of caller function will be applicable.** Hence, j in P will be accessed/updated according to caller function's environment and since, caller function i.e. main function here, does not have a local variable j, environment of function main for variable j would depend on the scoping used, But this scoping will be seen from the perspective of caller function i.e. function main, Not from the perspective of called function i.e. function P. So, it is like accessing variable j in the main function, not in P function. Hence, for this example 3, in both scoping, the main function will be using Global variable j.

Example 4 : What will be the output if "Call by name" parameter passing technique is used in case of static and dynamic scopings?

```
global int j=100, i = 300;

void Q(x) {
    print(x+10);
    print (x); }

void P(x) {
    print(x+10);
    Q(i);
    print (x);
}
main()
{
int i =500;
int j =10;
P(j);
}
```

Answer : Since "Call by name" parameter passing technique is used, we can re-write the program as following :

```
global int j=100, i = 300;

void Q(x) {
    print(i+10);  // this i refers to i in the caller function i.e. P function's environment
    print (i); } // this i refers to i in the caller function i.e. P function's environment

void P(x) {
    print(j+10);  // this j refers to j in the caller function i.e. main function's environment
    Q(i);
    print (j);    // this j refers to j in the caller function i.e. main function's environment
}
```

```
main()
{
int i =500;
int j =10;
P(j);
}
```

In case of Dynamic scoping : 20,510,500,10

In case of Static scoping : 20,310,300,10

In Static scoping, variable i in P function's environment refers to the global variable i. In Dynamic scoping, variable i in P function's environment refers to the main function's i.

> NOTE 2 (Technical Detail 2) :
> if any of the local variables in the called procedure clash with the caller's variables, they(called function's clashing variables) must be renamed uniquely before substitution.

Example 5 : What will be the output if "Call by name" parameter passing technique is used.

Given Program :

```
void P(x) {
    int j=100;
 print(x+10);
    print(j);
    print (x);
}
main()
{
int j = 10;
P(j);
}
```

Answer : Since "Call by name" parameter passing technique is used, we can re-write the program as following :

Caller function's Actual argument contains variable j which clashes with called function P's local variable j, hence, we rename called function P's local variable j and change it to j'.

```
void P(x) {
    int j'=100;
 print(j+10);   // this j refers to j in the caller function i.e. main function's environment
    print(j');    // this j' refers to the local variable j' in P.
    print (j);    // this j refers to j in the caller function i.e. main function's environment
}
main()
{
int j = 10;
P(j);
}
```

Hence, output : 20,100, 10.

Coming to the actual GATE question, we'll call it example 6.

Example 6 : What will be the output if "Call by name" parameter passing technique is used, in case of static and dynamic scopings?

```
global int i=100, j=5;
void P(x) {
 int i=10;
    print(x+10);
    i=200;
    j=20;
    print (x);
}
main() {P(i+j);}
```

Answer : Since "Call by name" parameter passing technique is used, we can re-write the program as following :

Caller function's Actual argument contains variable i which clashes with called function P's local variable i, hence, we rename called function P's local variable i and change it to i'.

```
global int i=100, j=5;
```

```
void P(x) {
 int i'=10;   // this i' refers to the local variable i' in function P.
    print(i+j+10);  // this i,j refers to i,j in the caller function i.e. main function's environment
    i'=200;    // this i' refers to the local variable i' in function P.
    j=20;      // this j refers to j in the caller function i.e. main function's environment
    print (i+j);   // this i,j refers to i,j in the caller function i.e. main function's environment
}
main() {P(i+j);}
```

In case of Static scoping : 115, 120

In case of Dynamic scoping : 115, 120

Note that there are no local variable i, j in main function, so, when we say that i, j refer to the i, j in main's environment , we mean that If i, j were accessed/updated in main function then depending on the scoping, which i, j would they refer.

Here, in this question, in both static and dynamic scoping case, i, j will refer to the Global variables.

And in function P, in the 4th statement (i.e. $j = 20$), the Global variable j will be updated.

Hence, No Option is correct for the actual above GATE question.

Example 7: What will be the output if "Call by name" parameter passing technique is used in case of static and dynamic scopings?

```
global int j=100, i = 300;

void Q(x) {
    print(x+10);
    print (x); }

void P(x) {
    int i = 400;
    int j = 600;
    print(x+10);
    Q(i);
    Q(j);
    print (x);
}
main()
{
int i =500;
Q(i);
int j =10;
P(j);
}
```

Answer :

in both scoping, for this question, output : 510,500,20,410,400,610,600,10.

Note that When main calls $P(j)$, then Caller function main's Actual argument contains variable j which clashes with called function P's local variable j, hence, we rename called function P's local variable j and change it to j'. So, in function P, the 5th statement becomes $Q(j')$.

Also note that when main calls Q(i), then x in Q is replaced with i. When P calls Q(i), then x in Q is replaced with i. When P calls Q(j'), then x in Q is replaced with j'.

Note that "Direct Substitution of actual parameters in the place of formal parameter in the called procedure" is only the Idea of Call-by-name, Not the actual practical implementation. Compiler does Not do Direct Substitution blindly. Pass-by-name is difficult to implement. Argument expressions must be compiled to special parameterless procedures called *thunks*. These thunks are passed into the called procedure and used whenever necessary to evaluate or re-evaluate the argument.

But we do not need to go into practical implementation details because the Idea remains same. Hence, we can solve all the questions using above idea of call-by-name.

> NOTE 3 (Technical Detail 3) :
> if any of the variables in the called procedure clash with the caller's variables, they(called function's clashing variables) must be renamed uniquely before substitution. Clashing variables need not be local variables of the called function.

But remember that when we rename a variable, we don't really rename it. We rename it just to eliminate the possibility of confusion. So, when we rename j to j', we must not forget that j' is actually j only in the first place.

The following example will illustrate this point :

Example 8: What will be the output if "Call by name" parameter passing technique is used in case of static and dynamic scopings?

```
global int j=100, i = 300;

void Q(x) {
    print(i);
    print(j);
    print(x+10);
    print (x); }

void P(x) {
    int i = 400;
    int j = 600;
    print(x+10);
    Q(i);
    Q(j);
    print (x);
}
main()
{
int i =500;
int j =10;
Q(i);
P(j);
}
```

Answer :

In case of Static scoping : 300, 100, 510, 500, 20, 300, 100, 410, 400, 300, 100, 610, 600, 10

In case of Dynamic scoping : 500, 10, 510, 500, 20, 400, 600, 410, 400, 400, 600, 610, 600, 10

```
global int j=100, i = 300;

void Q(x) {
    print(i); // this i refers to the Global i in case of static scoping and in case of dynamic scoping, acco
    print(j); // this j refers to the Global j in case of static scoping and in case of dynamic scoping, acco
    print(x+10);  // variables that are substituted here, refer to the corresponding variables in the caller f
    print (x); } // variables that are substituted here, refer to the corresponding variables in the caller fu

void P(x) {
    int i = 400;
    int j = 600;
    print(x+10); // variables that are substituted here, refer to the corresponding variables in the caller fu
    Q(i);
    Q(j);
    print (x); // variables that are substituted here, refer to the corresponding variables in the caller func
}
main()
{
int i =500;
int j =10;
Q(i);
P(j);
}
```

When main calls $Q(i)$, we substitute i in place of x and it(Q) becomes:

```
void Q(x) {
    print(i');  // This is renamed as i' and it refers to
                //global variable i in case of static scoping and
                //in case of dynamic scoping it refers to
                //variable i in main function.
    print(j);
    print(i+10);
    print (i);
}
```

Hence, the two different $i's$ should be distinguished properly.

---

https://www2.cs.sfu.ca/~cameron/Teaching/383/PassByName.html

https://www2.cs.arizona.edu/classes/cs520/spring06/06parameters.pdf

References

24 votes -- Deepak Poonia (23.3k points)

2.14.12 Parameter Passing: GATE CSE 2004 | Question: 2,ISRO2017-54 top https://gateoverflow.in/999

✔ ans (**D**).

Option A will not exchange the values of x and y because parameters are passed by value in C. i.e., the code is exchanging x' and y' which are having the values of x and y respectively.

Option B will not swap the value

```
void swap(int a, int b)
```

Here, it is wrong to pass in address (int*) as the parameters are of int type, even sizeof int and int* varies depending on the compiler. Now, even if ignoring this error, the given code would not exchange the values of x and y as it is merely exchanging p_1' and p_2' where p_1' and p_2' are containing the copies of the addresses of x and y respectively. (Even addresses are passed by value in C language).

Option C is false, return value is not required for exchanging the variables.

Option D is correct. We cannot use $swap(x, y)$ because parameters are passed by value. Only way now to exchange the variables are by passing their addresses and then modifying the contents using the de-referencing operator $(*)$.

28 votes -- minal (13.1k points)

2.14.13 Parameter Passing: GATE CSE 2016 Set 1 | Question: 36 top https://gateoverflow.in/39701

✔ It is a bit confusing as variable declaration is not explicit. But we can see that " $a = 3$" and "$a = 1$" are declaring new variables, one in global and other in local space.

Main is calling $m(a)$. Since there is no local 'a', 'a' here is the global one.

In m, we have "$a = 1$" which declares a local " a" and gives 1 to it. "$a = y - a$" assigns $3 - 1 = 2$ to 'a'.

Now, in $n(x)$, 'a' is used and as per dynamic scoping this ' a' comes from '$m()$' and not the global one. So, " $x = x * a$" assigns "$2 * 2 = 4$" to "x" and 4 is printed. Being passed by reference, "a" in $m()$ also get updated to 4. So, **D** is the answer here.

113 votes -- Arjun Suresh (330k points)

2.14.14 Parameter Passing: GATE IT 2007 | Question: 33 top https://gateoverflow.in/3466

✔ Correct answer is (**D**)

CALL BY VALUE :- i as global variable declared. Then in $main()$ a local variable j as integer declared i.e $j = 60$ And global variable i initialized to 50 by $i = 50$. Now procedure f called and values of i and j are passed to it. i.e., in $f(i, j) \rightarrow f(x, y)$ content of memory location of i (here 50) is copied to memory location of x (which is different from i) and content of memory location of j (here, 60) is copied to memory location of y. Then in $f(x, y)$ $i = 100$ changes the global i to 100, $X = 10$ changes the local X from 50 to 10 and $Y = y + i$ means $y = 60 + 100 = 160$. Now when return back to main, i and j will be 100 and 60 respectively.

CALL BY REFERENCE:- Now procedure f called and passed reference of i and j to it. i.e., in $f(i, j) \rightarrow f(x, y)$ x and y are new names (aliases) pointing to the same memory location of i and j respectively. So, $i = 100$ changes the global i to 100 and $x = 10$ means x as well as global $i = 10$ (as the i being passed is the global variable and x and i share the same address).
$y = y + i$ means $y = 60 + 10 = 70$ and this changes the value of j also to 70 as j and y have the same address. Now when return back to main, i and j will be 10 and 70 respectively.

44 votes -- Kalpna Bhargav (2.5k points)

2.15 Parsing (46) top

2.15.1 Parsing: GATE CSE 1987 | Question: 1-xiv top https://gateoverflow.in/80295

An operator precedence parser is a

A. Bottom-up parser.
B. Top-down parser.
C. Back tracking parser.

D. None of the above.

gate1987 compiler-design parsing

Answer

2.15.2 Parsing: GATE CSE 1988 | Question: 10ia top https://gateoverflow.in/94390

Consider the following grammar:

- $S \rightarrow S$
- $S \rightarrow SS \mid a \mid \epsilon$

Construct the collection of sets of $\text{LR}\ (0)$ items for this grammar and draw its goto graph.

gate1988 compiler-design descriptive grammar parsing

Answer

2.15.3 Parsing: GATE CSE 1988 | Question: 10ib top https://gateoverflow.in/94391

Consider the following grammar:

- $S \rightarrow S$
- $S \rightarrow SS \mid a \mid \epsilon$

Indicate the shift-reduce and reduce-reduce conflict (if any) in the various states of the $\text{LR}(0)$ parser.

gate1988 compiler-design descriptive grammar parsing

Answer

2.15.4 Parsing: GATE CSE 1989 | Question: 1-iii top https://gateoverflow.in/87046

Merging states with a common core may produce __________ conflicts and does not produce ___________ conflicts in an LALR parser.

gate1989 descriptive compiler-design parsing

Answer

2.15.5 Parsing: GATE CSE 1992 | Question: 02,xiii top https://gateoverflow.in/570

For a context free grammar, FOLLOW(A) is the set of terminals that can appear immediately to the right of non-terminal A in some "sentential" form. We define two sets LFOLLOW(A) and RFOLLOW(A) by replacing the word "sentential" by "left sentential" and "right most sentential" respectively in the definition of FOLLOW (A).

A. FOLLOW(A) and LFOLLOW(A) may be different.
B. FOLLOW(A) and RFOLLOW(A) are always the same.
C. All the three sets are identical.
D. All the three sets are different.

gate1992 parsing compiler-design normal multiple-selects

Answer

2.15.6 Parsing: GATE CSE 1992 | Question: 02,xiv top https://gateoverflow.in/571

Consider the $\text{SLR}(1)$ and $\text{LALR}\ (1)$ parsing tables for a context free grammar. Which of the following statement is/are true?

A. The *goto* part of both tables may be different.
B. The *shift* entries are identical in both the tables.
C. The *reduce* entries in the tables may be different.
D. The *error* entries in tables may be different

gate1992 compiler-design normal parsing multiple-selects

Answer

2.15.7 Parsing: GATE CSE 1993 | Question: 25 top

https://gateoverflow.in/2321

A simple Pascal like language has only three statements.

i. assignment statement e.g. x:=expression
ii. loop construct e.g. for i:=expression to expression do statement
iii. sequencing e.g. begin statement ;…; statement end

A. Write a context-free grammar (CFG) for statements in the above language. Assume that expression has already been defined. Do not use optional parenthesis and * operator in CFG.
B. Show the parse tree for the following statements:

```
for j:=2 to 10 do
begin
    x:=expr1;
    y:=expr2;
end
```

gate1993 compiler-design parsing normal descriptive

Answer

2.15.8 Parsing: GATE CSE 1995 | Question: 8 top

https://gateoverflow.in/2643

Construct the $\text{LL}(1)$ table for the following grammar.

1. $Expr \rightarrow \_Expr$
2. $Expr \rightarrow (Expr)$
3. $Expr \rightarrow Var\ ExprTail$
4. $ExprTail \rightarrow \_Expr$
5. $Expr \rightarrow \lambda$
6. $Var \rightarrow Id\ VarTail$
7. $VarTail \rightarrow (Expr)$
8. $VarTail \rightarrow \lambda$
9. $Goal \rightarrow Expr\$$

gate1995 compiler-design parsing normal descriptive

Answer

2.15.9 Parsing: GATE CSE 1998 | Question: 1.26 top

https://gateoverflow.in/1663

Which of the following statements is true?

A. SLR parser is more powerful than LALR
B. LALR parser is more powerful than Canonical LR parser
C. Canonical LR parser is more powerful than LALR parser
D. The parsers SLR, Canonical CR, and LALR have the same power

gate1998 compiler-design parsing normal

Answer

2.15.10 Parsing: GATE CSE 1998 | Question: 1.27 top

https://gateoverflow.in/1664

Type checking is normally done during

A. lexical analysis
B. syntax analysis
C. syntax directed translation
D. code optimization

gate1998 compiler-design parsing easy

Answer

2.15.11 Parsing: GATE CSE 1998 | Question: 22 top

https://gateoverflow.in/1737

A. An identifier in a programming language consists of up to six letters and digits of which the first character must be a letter. Derive a regular expression for the identifier.

B. Build an $LL(1)$ parsing table for the language defined by the $LL(1)$ grammar with productions

$\text{Program} \rightarrow \text{ begin } d \text{ semi } X \text{ end}$

$X \rightarrow d \text{ semi } X \mid sY$

$Y \rightarrow \text{ semi } sY \mid \epsilon$

gate1998 compiler-design parsing descriptive

Answer

2.15.12 Parsing: GATE CSE 1999 | Question: 1.17 top

https://gateoverflow.in/1470

Which of the following is the most powerful parsing method?

A. LL (1)
B. Canonical LR
C. SLR
D. LALR

gate1999 compiler-design parsing easy

Answer

2.15.13 Parsing: GATE CSE 2000 | Question: 1.19, UGCNET-Dec2013-II: 30 top

https://gateoverflow.in/642

Which of the following derivations does a top-down parser use while parsing an input string? The input is scanned from left to right.

A. Leftmost derivation
B. Leftmost derivation traced out in reverse
C. Rightmost derivation
D. Rightmost derivation traced out in reverse

gate2000-cse compiler-design parsing normal ugcnetdec2013ii

Answer

2.15.14 Parsing: GATE CSE 2001 | Question: 16 top

https://gateoverflow.in/757

Consider the following grammar with terminal alphabet $\Sigma = \{a, (,), +, *\}$ and start symbol E. The production rules of the grammar are:

- $E \rightarrow aA$
- $E \rightarrow (E)$
- $A \rightarrow +E$
- $A \rightarrow *E$
- $A \rightarrow \epsilon$

a. Compute the FIRST and FOLLOW sets for E and A.
b. Complete the LL(1) parse table for the grammar.

gate2001-cse compiler-design parsing normal descriptive

Answer

2.15.15 Parsing: GATE CSE 2002 | Question: 22 top

https://gateoverflow.in/875

A. Construct all the parse trees corresponding to $i + j * k$ for the grammar
   $E \rightarrow E + E$
   $E \rightarrow E * E$
   $E \rightarrow id$
B. In this grammar, what is the precedence of the two operators $*$ and $+$?
C. If only one parse tree is desired for any string in the same language, what changes are to be made so that the resulting LALR(1) grammar is unambiguous?

gate2002-cse compiler-design parsing normal descriptive

Answer

2.15.16 Parsing: GATE CSE 2003 | Question: 16 top

https://gateoverflow.in/906

Which of the following suffices to convert an arbitrary CFG to an LL(1) grammar?

A. Removing left recursion alone
B. Factoring the grammar alone
C. Removing left recursion and factoring the grammar
D. None of the above

gate2003-cse compiler-design parsing easy

Answer

2.15.17 Parsing: GATE CSE 2003 | Question: 17 top

https://gateoverflow.in/907

Assume that the SLR parser for a grammar G has n_1 states and the LALR parser for G has n_2 states. The relationship between n_1 and n_2 is

A. n_1 is necessarily less than n_2
B. n_1 is necessarily equal to n_2
C. n_1 is necessarily greater than n_2
D. None of the above

gate2003-cse compiler-design parsing easy

Answer

2.15.18 Parsing: GATE CSE 2003 | Question: 57 top

https://gateoverflow.in/945

Consider the grammar shown below.

- $S \rightarrow C\ C$
- $C \rightarrow c\ C \mid d$

This grammar is

A. LL(1)
B. SLR(1) but not LL(1)
C. LALR(1) but not SLR(1)
D. LR(I) but not LALR(1)

gate2003-cse compiler-design grammar parsing normal

Answer

2.15.19 Parsing: GATE CSE 2005 | Question: 14 top

https://gateoverflow.in/1350

The grammar $A \rightarrow AA \mid (A) \mid \epsilon$ is not suitable for predictive-parsing because the grammar is:

A. ambiguous
B. left-recursive
C. right-recursive
D. an operator-grammar

gate2005-cse compiler-design parsing grammar easy

Answer

2.15.20 Parsing: GATE CSE 2005 | Question: 60 top

https://gateoverflow.in/1383

Consider the grammar:

$$S \rightarrow (S) \mid a$$

Let the number of states in SLR (1), LR(1) and LALR(1) parsers for the grammar be n_1, n_2 and n_3 respectively. The following relationship holds good:

A. $n_1 < n_2 < n_3$
B. $n_1 = n_3 < n_2$
C. $n_1 = n_2 = n_3$
D. $n_1 \geq n_3 \geq n_2$

gate2005-cse compiler-design parsing normal

Answer

2.15.21 Parsing: GATE CSE 2005 | Question: 83a top

https://gateoverflow.in/1405

Statement for Linked Answer Questions 83a & 83b:

Consider the following expression grammar. The semantic rules for expression evaluation are stated next to each grammar production.

$$\begin{array}{r|l} E \rightarrow number & E.val = number.val \\ \mid E \text{ `}+\text{' } E & E^{(1)}.val = E^{(2)}.val + E^{(3)}.val \\ \mid E \text{ `}\times\text{' } E & E^{(1)}.val = E^{(2)}.val \times E^{(3)}.val \end{array}$$

The above grammar and the semantic rules are fed to a *yaac* tool (which is an LALR(1) parser generator) for parsing and evaluating arithmetic expressions. Which one of the following is true about the action of *yaac* for the given grammar?

A. It detects *recursion* and eliminates recursion
B. It detects *reduce-reduce* conflict, and resolves
C. It detects *shift-reduce* conflict, and resolves the conflict in favor of a *shift* over a *reduce* action
D. It detects *shift-reduce* conflict, and resolves the conflict in favor of a *reduce* over a *shift* action

gate2005-cse compiler-design parsing difficult

Answer

2.15.22 Parsing: GATE CSE 2005 | Question: 83b top

https://gateoverflow.in/87037

Consider the following expression grammar. The semantic rules for expression evaluation are stated next to each grammar production.

$$\begin{array}{r|l} E \rightarrow number & E.val = number.val \\ \mid E \text{ `}+\text{' } E & E^{(1)}.val = E^{(2)}.val + E^{(3)}.val \\ \mid E \text{ `}\times\text{' } E & E^{(1)}.val = E^{(2)}.val \times E^{(3)}.val \end{array}$$

Assume the conflicts of this question are resolved using yacc tool and an LALR(1) parser is generated for parsing arithmetic expressions as per the given grammar. Consider an expression $3 \times 2 + 1$. What precedence and associativity properties does the generated parser realize?

A. Equal precedence and left associativity; expression is evaluated to 7
B. Equal precedence and right associativity; expression is evaluated to 9
C. Precedence of '$\times$' is higher than that of '$+$', and both operators are left associative; expression is evaluated to 7
D. Precedence of '$+$' is higher than that of '$\times$', and both operators are left associative; expression is evaluated to 9

gate2005-cse compiler-design parsing normal

Answer

2.15.23 Parsing: GATE CSE 2006 | Question: 58 top

https://gateoverflow.in/1836

Consider the following grammar:

- $S \to FR$
- $R \to *S \mid \varepsilon$
- $F \to id$

In the predictive parser table M of the grammar the entries $M[S, id]$ and $M[R, \$]$ respectively are

A. $\{S \to FR\}$ and $\{R \to \varepsilon\}$
B. $\{S \to FR\}$ and $\{\}$
C. $\{S \to FR\}$ and $\{R \to *S\}$
D. $\{F \to id\}$ and $\{R \to \varepsilon\}$

gate2006-cse compiler-design parsing normal

Answer

2.15.24 Parsing: GATE CSE 2006 | Question: 7 top

https://gateoverflow.in/886

Consider the following grammar

- $S \to S * E$
- $S \to E$
- $E \to F + E$
- $E \to F$
- $F \to id$

Consider the following LR(0) items corresponding to the grammar above

i. $S \to S * .E$
ii. $E \to F. + E$
iii. $E \to F + .E$

Given the items above, which two of them will appear in the same set in the canonical sets-of-items for the grammar?

A. i and ii
B. ii and iii
C. i and iii
D. None of the above

gate2006-cse compiler-design parsing normal

Answer

2.15.25 Parsing: GATE CSE 2007 | Question: 18 top

https://gateoverflow.in/1216

Which one of the following is a top-down parser?

A. Recursive descent parser.
B. Operator precedence parser.
C. An LR(k) parser.
D. An LALR(k) parser.

gate2007-cse compiler-design parsing normal

Answer

2.15.26 Parsing: GATE CSE 2008 | Question: 11 top

https://gateoverflow.in/409

Which of the following describes a handle (as applicable to LR-parsing) appropriately?

A. It is the position in a sentential form where the next shift or reduce operation will occur
B. It is non-terminal whose production will be used for reduction in the next step
C. It is a production that may be used for reduction in a future step along with a position in the sentential form where the next shift or reduce operation will occur
D. It is the production p that will be used for reduction in the next step along with a position in the sentential form where the right hand side of the production may be found

gate2008-cse compiler-design parsing normal

Answer

2.15.27 Parsing: GATE CSE 2008 | Question: 55 top

https://gateoverflow.in/478

An LALR(1) parser for a grammar G can have shift-reduce (S-R) conflicts if and only if

A. The SLR(1) parser for G has S-R conflicts
B. The LR(1) parser for G has S-R conflicts
C. The LR(0) parser for G has S-R conflicts
D. The LALR(1) parser for G has reduce-reduce conflicts

gate2008-cse compiler-design parsing normal

Answer

2.15.28 Parsing: GATE CSE 2009 | Question: 42 top

https://gateoverflow.in/1328

Which of the following statements are TRUE?

I. There exist parsing algorithms for some programming languages whose complexities are less than $\Theta(n^3)$
II. A programming language which allows recursion can be implemented with static storage allocation.
III. No L-attributed definition can be evaluated in the framework of bottom-up parsing.
IV. Code improving transformations can be performed at both source language and intermediate code level.

A. I and II
B. I and IV
C. III and IV
D. I, III and IV

gate2009-cse compiler-design parsing normal

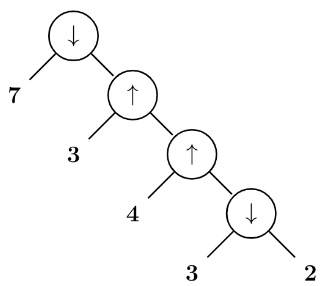
Answer

2.15.29 Parsing: GATE CSE 2011 | Question: 27 top

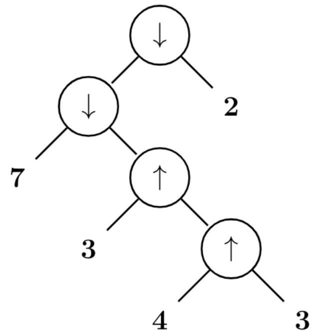
https://gateoverflow.in/2129

Consider two binary operators '$\uparrow$' and '$\downarrow$' with the precedence of operator $\downarrow$ being lower than that of the operator $\uparrow$. Operator $\uparrow$ is right associative while operator $\downarrow$ is left associative. Which one of the following represents the parse tree for expression $(7 \downarrow 3 \uparrow 4 \uparrow 3 \downarrow 2)$

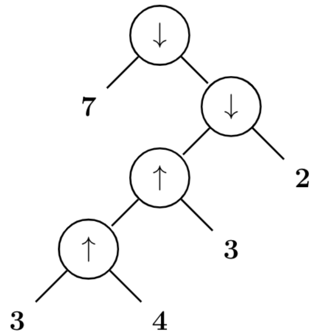
A.



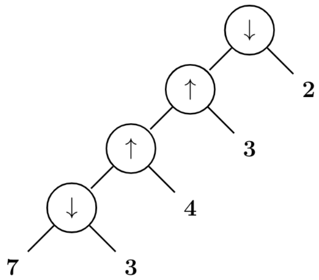
B.



C.



D.



gate2011-cse compiler-design parsing normal

Answer

2.15.30 Parsing: GATE CSE 2012 | Question: 52 top

https://gateoverflow.in/2181

For the grammar below, a partial $LL(1)$ parsing table is also presented along with the grammar. Entries that need to be filled are indicated as $E1, E2$, and $E3$. ε is the empty string, \$ indicates end of input, and, $|$ separates alternate right hand sides of productions.

- $S \rightarrow aAbB \mid bAaB \mid \varepsilon$
- $A \rightarrow S$
- $B \rightarrow S$

	a	b	\$
S	E1	E2	$S \rightarrow \varepsilon$
A	$A \rightarrow S$	$A \rightarrow S$	error
B	$B \rightarrow S$	$B \rightarrow S$	$E3$

The FIRST and FOLLOW sets for the non-terminals A and B are

A. $\text{FIRST}(A) = \{a, b, \varepsilon\} = \text{FIRST}(B)$
$\text{FOLLOW}(A) = \{a, b\}$
$\text{FOLLOW}(B) = \{a, b, \$\}$

B. $\text{FIRST}(A) = \{a, b, \$\}$
$\text{FIRST}(B) = \{a, b, \varepsilon\}$
$\text{FOLLOW}(A) = \{a, b\}$
$\text{FOLLOW}(B) = \{\$\}$

C. $\text{FIRST}(A) = \{a, b, \varepsilon\} = \text{FIRST}(B)$
$\text{FOLLOW}(A) = \{a, b\}$
$\text{FOLLOW}(B) = \varnothing$

D. $\text{FIRST}(A) = \{a, b\} = \text{FIRST}(B)$
$\text{FOLLOW}(A) = \{a, b\}$
$\text{FOLLOW}(B) = \{a, b\}$

gate2012-cse compiler-design parsing normal

Answer

2.15.31 Parsing: GATE CSE 2012 | Question: 53 top

☛ https://gateoverflow.in/43312

For the grammar below, a partial $LL(1)$ parsing table is also presented along with the grammar. Entries that need to be filled are indicated as $E1, E2$, and $E3$. ε is the empty string, $\$$ indicates end of input, and, $|$ separates alternate right hand sides of productions.

- $S \rightarrow aAbB \mid bAaB \mid \varepsilon$
- $A \rightarrow S$
- $B \rightarrow S$

	a	b	$
S	E1	E2	$S \rightarrow \varepsilon$
A	$A \rightarrow S$	$A \rightarrow S$	error
B	$B \rightarrow S$	$B \rightarrow S$	$E3$

The appropriate entries for $E1, E2$, and $E3$ are

A. $E1 : S \rightarrow aAbB, A \rightarrow S$
$E2 : S \rightarrow bAaB, B \rightarrow S$
$E1 : B \rightarrow S$

B. $E1 : S \rightarrow aAbB, S \rightarrow \varepsilon$
$E2 : S \rightarrow bAaB, S \rightarrow \varepsilon$
$E3 : S \rightarrow \varepsilon$

C. $E1 : S \rightarrow aAbB, S \rightarrow \varepsilon$
$E2 : S \rightarrow bAaB, S \rightarrow \varepsilon$
$E3 : B \rightarrow S$

D. $E1 : A \rightarrow S, S \rightarrow \varepsilon$
$E2 : B \rightarrow S, S \rightarrow \varepsilon$
$E3 : B \rightarrow S$

normal gate2012-cse compiler-design parsing

Answer

2.15.32 Parsing: GATE CSE 2013 | Question: 40 top

☛ https://gateoverflow.in/1551

Consider the following two sets of LR(1) items of an LR(1) grammar.

$$\begin{array}{r|l} X \rightarrow c.X, c\,/\,d & X \rightarrow c.X, \$ \\ X \rightarrow .cX, c\,/\,d & X \rightarrow .cX, \$ \\ X \rightarrow .d, c\,/\,d & X \rightarrow .d, \$ \end{array}$$

Which of the following statements related to merging of the two sets in the corresponding LALR parser is/are **FALSE**?

1. Cannot be merged since look aheads are different.
2. Can be merged but will result in S-R conflict.
3. Can be merged but will result in R-R conflict.
4. Cannot be merged since goto on c will lead to two different sets.

A. 1 only
B. 2 only
C. 1 and 4 only
D. 1, 2, 3 and 4

gate2013-cse compiler-design parsing normal

Answer

2.15.33 Parsing: GATE CSE 2013 | Question: 9 top https://gateoverflow.in/1418

What is the maximum number of reduce moves that can be taken by a bottom-up parser for a grammar with no epsilon and unit-production (i.e., of type $A \to \epsilon$ and $A \to a$) to parse a string with n tokens?

A. $n/2$
B. $n-1$
C. $2n-1$
D. 2^n

gate2013-cse compiler-design parsing normal

Answer

2.15.34 Parsing: GATE CSE 2014 Set 1 | Question: 34 top https://gateoverflow.in/1807

A canonical set of items is given below

$S \to L. > R$

$Q \to R.$

On input symbol $<$ the set has

A. a shift-reduce conflict and a reduce-reduce conflict.
B. a shift-reduce conflict but not a reduce-reduce conflict.
C. a reduce-reduce conflict but not a shift-reduce conflict.
D. neither a shift-reduce nor a reduce-reduce conflict.

gate2014-cse-set1 compiler-design parsing normal

Answer

2.15.35 Parsing: GATE CSE 2015 Set 3 | Question: 16 top https://gateoverflow.in/8413

Among simple LR (SLR), canonical LR, and look-ahead LR (LALR), which of the following pairs identify the method that is very easy to implement and the method that is the most powerful, in that order?

A. SLR, LALR
B. Canonical LR, LALR
C. SLR, canonical LR
D. LALR, canonical LR

gate2015-cse-set3 compiler-design parsing normal

Answer

2.15.36 Parsing: GATE CSE 2015 Set 3 | Question: 31 top https://gateoverflow.in/8488

Consider the following grammar G

$S \to F \mid H$

$F \to p \mid c$

$H \to d \mid c$

Where S, F, and H are non-terminal symbols, p, d, and c are terminal symbols. Which of the following statement(s) is/are correct?

S1: LL(1) can parse all strings that are generated using grammar G

S2: LR(1) can parse all strings that are generated using grammar G

A. Only S1
B. Only S2
C. Both S1 and S2
D. Neither S1 and S2

gate2015-cse-set3 compiler-design parsing normal

Answer

2.15.37 Parsing: GATE CSE 2016 Set 1 | Question: 45 top

https://gateoverflow.in/39697

The attribute of three arithmetic operators in some programming language are given below.

OPERATOR	PRECEDENCE	ASSOCIATIVITY	ARITY
$+$	High	Left	Binary
$-$	Medium	Right	Binary
$*$	Low	Left	Binary

The value of the expression $2-5+1-7*3$ in this language is ________.

gate2016-cse-set1 compiler-design parsing normal numerical-answers

Answer

2.15.38 Parsing: GATE CSE 2017 Set 1 | Question: 17 top

https://gateoverflow.in/118297

Consider the following grammar:

- $P \rightarrow xQRS$
- $Q \rightarrow yz \mid z$
- $R \rightarrow w \mid \varepsilon$
- $S \rightarrow y$

What is FOLLOW(Q)?

A. $\{R\}$
B. $\{w\}$
C. $\{w, y\}$
D. $\{w, \$\}$

gate2017-cse-set1 compiler-design parsing

Answer

2.15.39 Parsing: GATE CSE 2017 Set 1 | Question: 43 top

https://gateoverflow.in/118326

Consider the following grammar:

- stmt $\rightarrow$ **if** expr **then** expr **else** expr; stmt | Ò
- expr $\rightarrow$ term **relop** term | term
- term $\rightarrow$ id | number
- id $\rightarrow$ **a** | **b** | **c**
- number $\rightarrow$ $[0-9]$

where **relop** is a relational operator $(\text{e.g.}, <, >, \ldots)$, Òrefers to the empty statement, and **if**, **then**, **else** are terminals.
Consider a program P following the above grammar containing ten **if** terminals. The number of control flow paths in P is________ . For example. the program
if e_1 **then** e_2 **else** e_3
has 2 control flow paths. $e_1 \rightarrow e_2$ and $e_1 \rightarrow e_3$.

gate2017-cse-set1 compiler-design parsing normal numerical-answers

Answer

2.15.40 Parsing: GATE CSE 2017 Set 2 | Question: 6 top https://gateoverflow.in/118343

Which of the following statements about parser is/are CORRECT?

I. Canonical LR is more powerful than SLR
II. SLR is more powerful than LALR
III. SLR is more powerful than Canonical LR

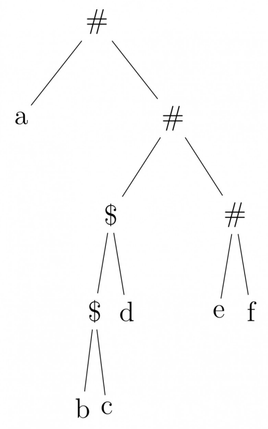
A. I only
B. II only
C. III only
D. II and III only

gate2017-cse-set2 compiler-design parsing

Answer

2.15.41 Parsing: GATE CSE 2018 | Question: 38 top https://gateoverflow.in/204112

Consider the following parse tree for the expression a#b cd#e#f, involving two binary operators $ and #.



Which one of the following is correct for the given parse tree?

A. $ has higher precedence and is left associative; # is right associative
B. # has higher precedence and is left associative; $ is right associative
C. $ has higher precedence and is left associative; # is left associative
D. $ has higher precedence and is right associative; # is left associative

gate2018-cse compiler-design parsing normal

Answer

2.15.42 Parsing: GATE CSE 2019 | Question: 19 top https://gateoverflow.in/302829

Consider the grammar given below:

- $S \rightarrow Aa$
- $A \rightarrow BD$
- $B \rightarrow b \mid \epsilon$
- $D \rightarrow d \mid \epsilon$

Let a, b, d and $\$$ be indexed as follows:

a	b	d	$\$$
3	2	1	0

Compute the FOLLOW set of the non-terminal B and write the index values for the symbols in the FOLLOW set in the descending order.(For example, if the FOLLOW set is $(a, b, d, \$)$, then the answer should be 3210)

gate2019-cse numerical-answers compiler-design parsing

Answer

2.15.43 Parsing: GATE CSE 2019 | Question: 3 top

https://gateoverflow.in/302845

Which one of the following kinds of derivation is used by LR parsers?

A. Leftmost
B. Leftmost in reverse
C. Rightmost
D. Rightmost in reverse

gate2019-cse compiler-design parsing

Answer

2.15.44 Parsing: GATE IT 2005 | Question: 83a top

https://gateoverflow.in/3849

Consider the context-free grammar

$E \rightarrow E + E$
$E \rightarrow (E * E)$
$E \rightarrow id$

where E is the starting symbol, the set of terminals is $\{id, (, +,), *\}$, and the set of nonterminals is $\{E\}$.

Which of the following terminal strings has more than one parse tree when parsed according to the above grammar?

A. $id + id + id + id$
B. $id + (id * (id * id))$
C. $(id * (id * id)) + id$
D. $((id * id + id) * id)$

gate2005-it compiler-design grammar parsing easy

Answer

2.15.45 Parsing: GATE IT 2005 | Question: 83b top

https://gateoverflow.in/3850

Consider the context-free grammar

- $E \rightarrow E + E$
- $E \rightarrow (E * E)$
- $E \rightarrow \text{id}$

where E is the starting symbol, the set of terminals is $\{id, (, +,), *\}$, and the set of non-terminals is $\{E\}$.

For the terminal string $id + id + id + id$, how many parse trees are possible?

A. 5
B. 4
C. 3
D. 2

gate2005-it compiler-design parsing normal

Answer

2.15.46 Parsing: GATE IT 2008 | Question: 79 top

https://gateoverflow.in/3393

A CFG G is given with the following productions where S is the start symbol, A is a non-terminal and a and b are terminals.

- $S \rightarrow aS \mid A$
- $A \rightarrow aAb \mid bAa \mid \epsilon$

For the string "$aabbaab$" how many steps are required to derive the string and how many parse trees are there?

A. 6 and 1
B. 6 and 2
C. 7 and 2
D. 4 and 2

gate2008-it compiler-design context-free-languages parsing normal

Answer

Answers: Parsing

2.15.1 Parsing: GATE CSE 1987 | Question: 1-xiv top

https://gateoverflow.in/80295

✔ **A. Bottom-up parser.**

An operator-precedence parser is a simple shift-reduce parser that is capable of parsing a subset of $\text{LR}(1)$ grammars. More precisely, the operator-precedence parser can parse all $\text{LR}(1)$ grammars where two consecutive non-terminals and epsilon never appear in the right-hand side of any rule.

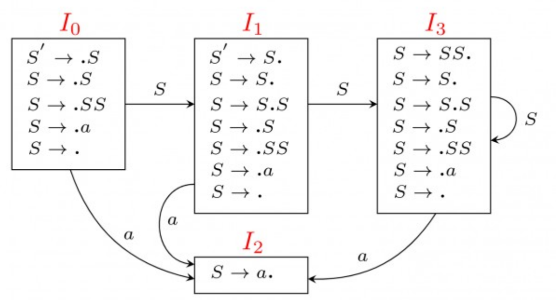
14 votes -- Soumya Jain (12.5k points)

2.15.2 Parsing: GATE CSE 1988 | Question: 10ia top

https://gateoverflow.in/94390

✔ The augmented production is $S' \rightarrow S$.

GOTO Graph:



Here, each of I_0, I_1, I_2, I_3 is a set of $LR(0)$ items. And hence I_0, I_1, I_2, I_3 are the collection of sets of $LR(0)$ items.

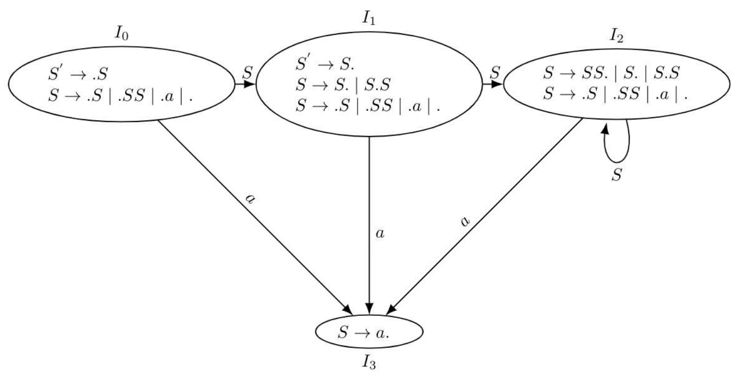
9 votes -- Sukanya Das (9.9k points)

2.15.3 Parsing: GATE CSE 1988 | Question: 10ib top

https://gateoverflow.in/94391

✔ The LR(0) DFA is shown below with states labeled as I_0, I_1, I_2, I_3.

Note : In the DFA, the transition $S \rightarrow \varepsilon$ is same as the transition $S \rightarrow .$ (as it is a reduce move).



The SR conflicts in this DFA are:

1. State I_0: Shift move on 'a' and a reduce move $S \rightarrow .$
2. State I_1: Shift move on 'a' and a reduce move $S \rightarrow .$
3. State I_2: Shift move on 'a' and a reduce move $S \rightarrow .$

(moves on non-terminal symbol 'S' are goto moves and not shift moves).

The RR conflicts in this DFA are:

1. State I_1: $S \rightarrow .$ and $S \rightarrow S.$
2. State I_2: ($S \rightarrow S.$ and $S \rightarrow .$) , ($S \rightarrow SS.$ and $S \rightarrow .$)

1 votes -- haralk10 (1.8k points)

2.15.4 Parsing: GATE CSE 1989 | Question: 1-iii top

https://gateoverflow.in/87046

✔ Merging states with a common core may produce **Reduce - Reduce** conflicts and does not produce **Shift- Reduce** conflicts in an LALR parser.

Refer : Link

References

16 votes -- Prashant Singh (47.1k points)

2.15.5 Parsing: GATE CSE 1992 | Question: 02,xiii top

https://gateoverflow.in/570

✔ **Ans - A,B,** LFollow may be different but RFollow and Follow will be same

Consider a Grammar -

$S \rightarrow AB$

$A \rightarrow a$

$B \rightarrow b$

Now only string derivable is $\{ab\}$.

Let's find Follow(A) in all cases :

i. **Follow(A)** - set of terminals that can appear immediately to the right of non-terminal A in some "sentential " form
$S \rightarrow AB \rightarrow Ab \rightarrow ab$

   Here, we notice only ' b' can appear to the right of A.

   Follow $(A) = \{b\}$

---

i. **LFollow(A)** - set of terminals that can appear immediately to the right of non-terminal A in some "left sentential" form
$S \rightarrow AB \rightarrow aB \rightarrow ab$

   Here, we notice no terminal can appear to the right of A.

   LFollow $(A) = \{\}$

---

i. **RFollow(A)** - set of terminals that can appear immediately to the right of non-terminal A in some "right most sentential" form
$S \rightarrow AB \rightarrow Ab \rightarrow ab$

   Here, we notice only ' b' can appear to the right of A.

   RFollow $(A) = \{b\}$

58 votes -- Himanshu Agarwal (12.4k points)

2.15.6 Parsing: GATE CSE 1992 | Question: 02,xiv top

https://gateoverflow.in/571

✔

- Goto part & shift entry must be same.
- Reduce entry & error entry may b different due to conflicts.

Correct Answer: B;C;D.

35 votes -- Digvijay (44.9k points)

2.15.7 Parsing: GATE CSE 1993 | Question: 25 top

https://gateoverflow.in/2321

- $P \rightarrow A \mid L \mid A \mid S \mid \epsilon$
- $A \rightarrow V := E$
- $L \rightarrow for\ V := E\ to\ E\ do\ P$
- $S \rightarrow begin\ P\ P\ end \mid \epsilon$
- $V \rightarrow id$

5 votes -- Arjun Suresh (330k points)

2.15.8 Parsing: GATE CSE 1995 | Question: 8 top

https://gateoverflow.in/2643

Non-Terminals: $\{Goal\ , Expr\ , ExprTail\ , Var\ , VarTail\ \}$

Terminals : $\{\_, (,), Id, \lambda\}$

Starting symbol : $\{Goal\}$ ($\because$ The end delimiter $\$$ is at the end of it in RHS)

Rewriting the grammar

1. $Goal \rightarrow Expr\$$
2. $Expr \rightarrow \_Expr \mid (Expr) \mid Var\ ExprTail \mid \lambda$
3. $ExprTail \rightarrow \_Expr$
4. $Var \rightarrow Id\ VarTail$
5. $VarTail \rightarrow (Expr) \mid \lambda$

Non-terminal	**First**	**Follow**
$Goal$	$\{\_, (, Id, \lambda\}$	$\{\$\}$
$Expr$	$\{\_, (, Id, \lambda\}$	$\{\$,)\ \}$
$ExprTail$	$\{\_\}$	$\{\$,)\ \}$
Var	$\{\ Id\}$	$\{\_\}$
$VarTail$	$\{(, \lambda\}$	$\{\_\}$

The $\text{LL}(1)$ table will be as follows:

	_	(	)	Id	λ	$
Goal	$Goal \rightarrow Expr\$$	$Goal \rightarrow Expr\$$		$Goal \rightarrow Expr\$$	$Goal \rightarrow Expr\$$	
Expr	$Expr \rightarrow \_Expr$	$Expr \rightarrow\ (Expr)$		$Expr \rightarrow Var\ ExprTail$	$Expr \rightarrow \lambda$	
ExprTail	$ExprTail \rightarrow \_Expr$					
Var				$Var \rightarrow Id\ VarTail$		
VarTail		$VarTail \rightarrow (Expr)$			$VarTail \rightarrow \lambda$	

4 votes -- Satbir Singh (21k points)

2.15.9 Parsing: GATE CSE 1998 | Question: 1.26 top

https://gateoverflow.in/1663

A. SLR paper is more powerful than LALR . **False .**
B. LALR parser is more powerful than Canonical LR parser . **False .**
C. Canonical LR parser is more powerful than LALR parser. **True.**
D. The parsers SLR, Canonical CR, and LALR have the same power. **False.**

answer - **C**

25 votes -- Ankit Rokde (6.9k points)

2.15.10 Parsing: GATE CSE 1998 | Question: 1.27 top

https://gateoverflow.in/1664

The answer is **c** .

The use of syntax analyser is used to create parse Tree. But along with Grammar as input to Syntax Analyser we add even semantic rules which form the basis of Syntax Directed Translation That help us in Evaluation of Expression .Remember that

Syntax Directed Translation is used in following cases

1. Conversion of infix to Postfix
2. Calculation of infix expression
3. For creating a Acyclic graph
4. Type Checking
5. Conversion of Binary number to Decimal
6. Counting the numbers of bits (0 or 1) in a binary number
7. Creation of syntax tree
8. To generate Intermediate code
9. Storing the data into Symbol table

70 votes -- spriti1991 (1.5k points)

2.15.11 Parsing: GATE CSE 1998 | Question: 22 top https://gateoverflow.in/1737

✔ **a.** $(letter)(letter + digit + \epsilon)^5$

b.

A. Program $\rightarrow$ begin d semi X end
B. $X \rightarrow d\ semiX$
C. $X \rightarrow sY$
D. $Y \rightarrow semi\ sY$
E. $Y \rightarrow \epsilon$

Variable	First	Follow
Program	begin	$
X	d, s	end
Y	semi, ϵ	end

Here, First(Y) contains ϵ so we need to add $Y \rightarrow \epsilon$ to Follow(Y)

Variable	begin	d	semi	s	end	$
Program	A					
X		B		C		
Y			D		Y $\rightarrow \epsilon$	

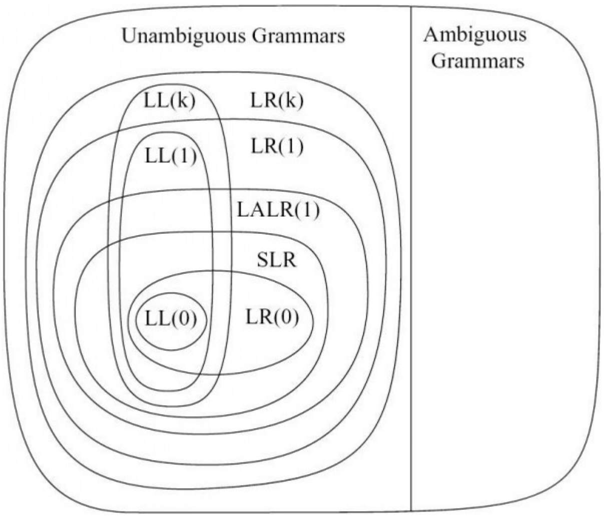
23 votes -- papesh (18k points)

2.15.12 Parsing: GATE CSE 1999 | Question: 1.17 top https://gateoverflow.in/1470

✔ Canonical LR is most powerful method

LR > LALR > SLR

so ans is **B**



27 votes

-- Pooja Palod (24.1k points)

2.15.13 Parsing: GATE CSE 2000 | Question: 1.19, UGCNET-Dec2013-II: 30 top

https://gateoverflow.in/642

✔ Top-down parser - Leftmost derivation

Bottom-Up parser - Reverse of rightmost derivation

21 votes

-- Vicky rix (7k points)

2.15.14 Parsing: GATE CSE 2001 | Question: 16 top

https://gateoverflow.in/757

✔ First $(E) = \{a, (\}$
First $(A) = \{+, *, \epsilon\}$
Follow $(E) =$ Follow $(A) = \{\ \$\ ,)\}$
LL(1) Parsing Table:

	a	(	)	+	*	$
E	E→ aA	E→ (E)				
A			$A \to \epsilon$	A→ +E	$A \to *$E	$A \to \epsilon$

32 votes

-- Aditya Gaurav (2.4k points)

2.15.15 Parsing: GATE CSE 2002 | Question: 22 top

https://gateoverflow.in/875

✔

a. two parse tree for i+j*k.
b. $+$ and $*$ having same precedence..
c. to make grammar LALR compatible give priority to $+$ over $*$ or vice versa.

following grammar is LALR(1)

$$\begin{aligned} E &\to E + T \\ &\quad |T \\ T &\to T * F \\ &\quad |F \\ F &\to id \end{aligned}$$

33 votes

-- Digvijay (44.9k points)

2.15.16 Parsing: GATE CSE 2003 | Question: 16 top

https://gateoverflow.in/906

✔ LL(1) parser is top down parser.

For top down parsers, the grammar should be unambiguous, deterministic and should not be left recursive.

All the 3 conditions must be satisfied for LL(1) parsers.

Now, even if all 3 conditions are satisfied we cannot get an LL(1) or even LL(k) (for any k) grammar for even a DCFG. This is because there are DCFLs which does not have an LL(k) grammar (see ref below). On the other hand for any DCFL, we can always have an LR(1) grammar.

http://mathoverflow.net/questions/31733/can-i-have-an-ll-grammar-for-every-deterministic-context-free-language

So, option **D** is correct.

References

63 votes -- Monanshi Jain (7k points)

2.15.17 Parsing: GATE CSE 2003 | Question: 17 top https://gateoverflow.in/907

✔ no of states in SLR and LALR are equal

and no of states in SLR and LALR are less than or equal to LR(1)

Correct Answer: B

27 votes -- Pooja Palod (24.1k points)

2.15.18 Parsing: GATE CSE 2003 | Question: 57 top https://gateoverflow.in/945

✔ ans is **A**

$First(S) = First(C) = \{c, d\}$

There are no multiple entries in single row of parsing table hence grammar is LL1

Note : If we have $A \rightarrow B \mid C$, for grammar to be LL(1) first(B) intersection First(C) should be null otherwise grammar is not LL1. If First(B) contains ϵ then Follow(A) intersection First(C) should be null. Using this we can say grammar is LL(1) or not without constructing parsing table.

An ϵ free LL(1) grammar is also SLR(1) and hence LALR(1) and LR(1) too.

62 votes -- Pooja Palod (24.1k points)

2.15.19 Parsing: GATE CSE 2005 | Question: 14 top https://gateoverflow.in/1350

✔ both **A** and **B** can be answers but **A** is a better answer. Because we have standard procedure for removing left-recursion but ambiguity is not easy to remove. - checking if a given CFG is ambiguous is a undecidable problem.

58 votes -- Vikrant Singh (11.2k points)

2.15.20 Parsing: GATE CSE 2005 | Question: 60 top https://gateoverflow.in/1383

✔ ans **B**

Both in SLR(1) and LALR(1), states are the LR(0) items while in LR(1) the states are LR(1) set of items. Number of LR(0) items can never be greater than number of LR(1) items. So, $n_1 = n_3 \leq n_2$, **B** choice. If we construct the states for the grammar we can replace $\leq$ with $<$.

31 votes -- Aditi Dan (4k points)

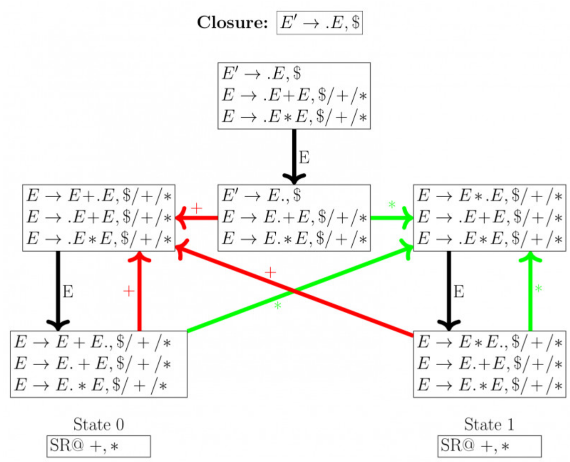
2.15.21 Parsing: GATE CSE 2005 | Question: 83a top https://gateoverflow.in/1405

✔ Given grammar:

$E \rightarrow \text{num}$

$E \rightarrow E + E \mid E * E$

First $LR(1)$ item : $E' \rightarrow \bullet E$,\$



YACC default action on SR: Choose SHIFT action

While parsing $3 * 2 + 1$, at some point of time stack content : $\begin{array}{|c|}\hline 1 \\ + \\ 2 \\ * \\ 3 \\ \hline\end{array}$

Then reduce handles one by one to generate output $= 9$.

- num does not create any conflict.
- Additionally here no states differ by lookahead symbols only.
- $\Rightarrow \text{LALR}(1)$ and $\text{LR}(1)$ tables are same.
- $LR(1)$ table only for state0 and state1:

	+	*	$
0	SR	SR	R
1	SR	SR	R

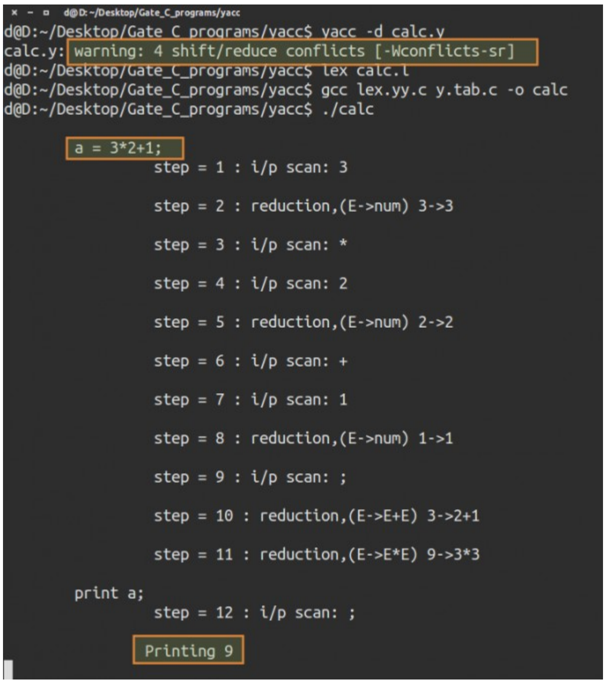
So total $2 + 2 = 4$ SR conflict originated in two states of the DFA.

- **Shift-reduce conflict:** *Yacc's* default action in the case of a shift-reduce conflict is to choose the shift action.
- **Reduce-reduce conflict :** *Yacc's* default action in the case of a reduce-reduce conflict is to reduce using the production that comes first, textually, in the input grammar specification.

---

and LEX-YACC-gcc output after implementing the given grammar :

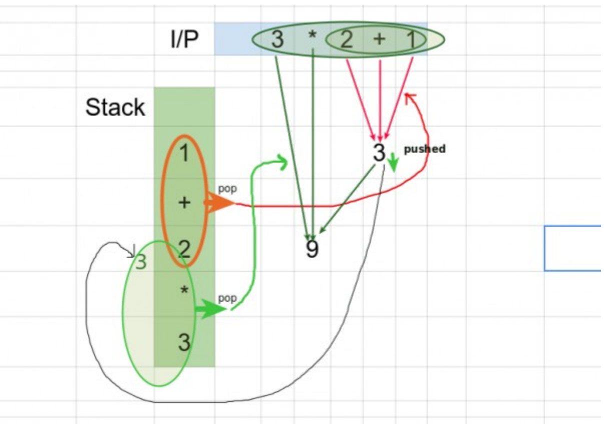
```
exp  : number        {$$ = $1;     printf("\t\t step = %d : reduction,(E->num) %d->%d\n\n",++step,$$,$1);}
     | exp '+' exp   {$$ = $1 + $3; printf("\t\t step = %d : reduction,(E->E+E) %d->%d+%d\n\n",++step,$$,$1,$3);}
     | exp '*' exp   {$$ = $1 * $3; printf("\t\t step = %d : reduction,(E->E*E) %d->%d*%d\n\n",++step,$$,$1,$3);}
     | identifier    {$$ = symbolVal($1);}
     ;

%%                /* C code */
```



As we can see from the output reduction on $E \to \text{num}$ is carried out as soon as top of stack contains a num. So, no conflict related to $E \to num$.

one example : Because of YACC shift preference, even if $3 * 2$ $(E * E)$ handle found on top of the stack at some point of time, it will shift on reading $+$ instead of reducing with $E \to E * E$. In this way, the complete input will be pushed into the stack. After that only reduce work starts as shown below.

- **Equal precedence** because of the given grammar $E \to E + E \mid E * E$, (single level)
- and **Right associativity** :



How YACC handles conflicts

Here are the ***required files*** (calc.l and calc.y) to regenerate the above interpreter.

Correct Answer: C

References

131 votes

-- Debashish Deka (40.7k points)

2.15.22 Parsing: GATE CSE 2005 | Question: 83b top

https://gateoverflow.in/87037

✔ LALR Parser is type of **Bottom up Parser** which uses **Right most Derivation**

For $3 \times 2 + 1$

$E \to E * E$ (Both shift and reduce possible but yacc prefers shift)

$\rightarrow E * E + E$
$\rightarrow E * E + 1$
$\rightarrow E * 2 + 1$
$\rightarrow E * 3$
$\rightarrow 3 * 3$
$\rightarrow 9$

All the productions are in same level therefore all have same precedence

Therefore Ans is **B. Equal precedence and right associativity; expression is evaluated to 9.**

44 votes -- Prajwal Bhat (7.6k points)

2.15.23 Parsing: GATE CSE 2006 | Question: 58 top

https://gateoverflow.in/1836

✔ First $S = \{id\}$
Follow $R = \{\$\}$

so $M[S, id] = S \rightarrow FR$
$M[R, \$] = R \rightarrow \epsilon$

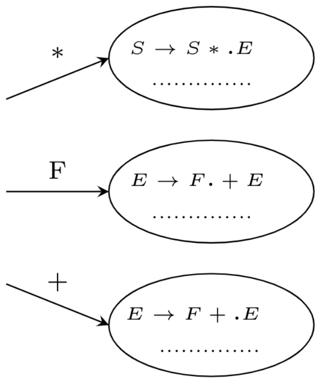
So ans is A

28 votes -- Pooja Palod (24.1k points)

2.15.24 Parsing: GATE CSE 2006 | Question: 7 top

https://gateoverflow.in/886

✔



$\Rightarrow$ NOT possible for these three items to be in same state

Correct Answer: D

29 votes -- Debashish Deka (40.7k points)

2.15.25 Parsing: GATE CSE 2007 | Question: 18 top

https://gateoverflow.in/1216

✔

1. Recursive descent parser-TOP DOWN PARSER
2. Operator precedence parser-BOTTOM UP PARSER
3. An LR(k) parser.-BOTTOM UP PARSER
4. An LALR(k) parser-BOTTOM UP PARSER

15 votes -- vnc (2.1k points)

2.15.26 Parsing: GATE CSE 2008 | Question: 11 top

https://gateoverflow.in/409

✔ A ***sentential form*** is the start symbol S of a grammar or any string in $(V \cup T)*$ that can be derived from S.

Consider the linear grammar

$(\{S, B\}, \{a, b\}, S, \{S \rightarrow aS, S \rightarrow B, B \rightarrow bB, B \rightarrow \lambda\})$.

A derivation using this grammar might look like this:

$$S \Rightarrow aS \Rightarrow aB \Rightarrow abB \Rightarrow abbB \Rightarrow abb$$

Each of $\{S, aS, aB, abB, abbB, abb\}$ is a sentential form.

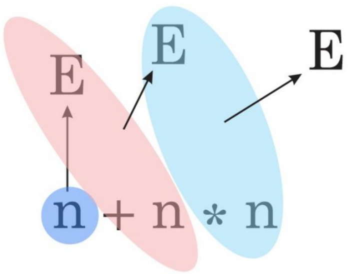
Because this grammar is linear, each sentential form has at most one variable. Hence there is never any choice about which variable to expand next.

Here, in *option D* the sentential forms are same but generated differently coz we are using here Bottom Up production.

Handle:
for example the grammar is:

$$\begin{aligned} E &\to E + n \\ E &\to E * n \\ E &\to n \end{aligned}$$

Then say to derive string $n + n * n$:



these are three different handles shown in 3 different colors = $\{n, E + n, E * n\}$

that's what **option D** says

69 votes -- Amar Vashishth (25.2k points)

2.15.27 Parsing: GATE CSE 2008 | Question: 55 top https://gateoverflow.in/478

Both LALR(1) and LR(1) parser uses LR(1) set of items to form their parsing tables. And LALR(1) states can be find by merging LR(1) states of LR(1) parser that have the same set of first components of their items.

i.e. if LR(1) parser has 2 states I and J with items $A \to a.bP, x$ and $A \to a.bP, y$ respectively, where x and y are look ahead symbols, then as these items are same with respect to their first component, they can be merged together and form one single state, let's say K. Here we have to take union of look ahead symbols. After merging, State K will have one single item as $A \to a.bP, x, y$. This way LALR(1) states are formed (i.e. after merging the states of LR(1)).

Now, $S - R$ conflict in LR(1) items can be there whenever a state has items of the form :

```
A-> a.bB , p
C-> d. , b

i.e. it is getting both shift and reduce at symbol b,
hence a conflict.
```

Now, as LALR(1) have items similar to LR(1) in terms of their first component, shift-reduce form will only take place if it is already there in LR(1) states. If there is no S-R conflict in LR(1) state it will never be reflected in the LALR(1) state obtained by combining LR(1) states.

Correct Answer: B

42 votes -- Madhab Paul Choudhury (2.8k points)

2.15.28 Parsing: GATE CSE 2009 | Question: 42 top https://gateoverflow.in/1328

Answer is **B**.

A. Yes there does exist parsing algorithms which run in $\Theta(n^3)$.
B. It cannot be implemented with static storage allocation. It needs dynamic memory allocation.
C. Every S-attributed definition is also an L-attributed definition and can be evaluated in the framework of bottom up parsing.
D. True.

38 votes -- Gate Keeda (15.9k points)

2.15.29 Parsing: GATE CSE 2011 | Question: 27 top https://gateoverflow.in/2129

✔ Answer is **B.**

To make the parse tree start compiling the identifiers into blocks based on associativity and precedence.

Grouping: $(7 \downarrow (3 \uparrow (4 \uparrow 3))) \downarrow 2$

Tree can be made by opening inner braces and move towards braces.

44 votes -- Sona Praneeth Akula (3.4k points)

2.15.30 Parsing: GATE CSE 2012 | Question: 52 top https://gateoverflow.in/2181

✔

- $\text{First}(S) = \text{First}(A) = \text{First}(B) = \{a, b, \epsilon\}$
- $\text{Follow}(A) = \{a, b\}$
- $\text{Follow}(B) = \text{Follow}(S) = \{a, b, \$\}$

So, the answer to question 52 is option A.

23 votes -- Pooja Palod (24.1k points)

2.15.31 Parsing: GATE CSE 2012 | Question: 53 top https://gateoverflow.in/43312

✔ To make $LL(1)$ parsing table first we have to find FIRST and FOLLOW sets from the given grammar.

- $\text{FIRST}(S) = \{a, b, \epsilon\}$
- $\text{FIRST}(A) = \{a, b, \epsilon\}$
- $\text{FIRST}(B) = \{a, b, \epsilon\}$
- $\text{FOLLOW}(S) = \{a, b, \$\}$
- $\text{FOLLOW}(A) = \{a, b\}$
- $\text{FOLLOW}(B) = \{a, b, \$\}$

Now lets make $LL(1)$ parse table

Non Terminal	a	b	$
S	$S \to aAbB$, $S \to \epsilon$	$S \to bAbB$, $S \to \epsilon$	$S \to \varepsilon$
A	$A \to S$	$A \to S$	
B	$B \to S$	$B \to S$	$B \to S$

Here is the explanation of entries asked in question

1. For E1 and E2 Look into $\text{FIRST}(S) = \{a, b, \epsilon\}$.
   a is because of $S \to aAbB$ and b is because of $B \to bAaB$
   So $M[S, a]$ and $M[S, b]$ will contain $S \to aAbB$ and $B \to bAaB$ respectively. For epsilon Look into $\text{FOLLOW}(S) = \{a, b, \$\}$. So $S \to \epsilon$ will be in $M[S, a]$, $M[S, b]$ and $M[S, \$]$
2. Now for E2 look into $\text{FIRST}(B) = \{a, b, \$\}$. a and b are because of $B \to S$.
   So $M[B, a]$ and $M[B.b]$ will contain $B \to S$ and for epsilon look into $\text{FOLLOW}(B) = \{a, b, \$\}$. Hence $M[B, \$]$ will contain $B \to S$

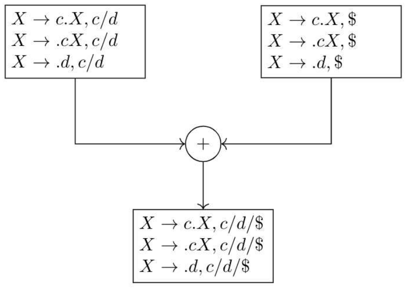
Now we get the answer as E1 is $S \to aAbB$, $S \to \epsilon$, E2 is $S \to bAaB$, $S \to \epsilon$ and E3 is $B \to S$.

Hence, **Option (C)** is correct.

34 votes -- Ashwani Kumar (13k points)

2.15.32 Parsing: GATE CSE 2013 | Question: 40 top https://gateoverflow.in/1551

✔



The TRUE statements are about merging of two states for $\text{LALR}(1)$ parser from $\text{LR}(1)$ parser.

1. The given two states can be merged because kernel of these are same, look aheads don't matter in merging.
2. The two states do not contain shift reduce conflict, so after merging the merged states cannot contain any S-R conflict.
3. There is no final item in both states, so no R-R conflict.
4. Merging of states does not depend on further GOTO part on any terminal.

Therefore, all the given statements in question are FALSE.

Option (**D**) is correct.

👍 49 votes -- Viral Kapoor (1.9k points)

2.15.33 Parsing: GATE CSE 2013 | Question: 9 top https://gateoverflow.in/1418

✔ Ans will be **B**

- $A \to BC$
- $B \to aa$
- $C \to bb$

Now suppose string is $aabb$. Then

$A \to BC$ (reduction 3)
$\to aaC$ (reduction 2)
$\to aabb$ (reduction 1)

$n = 4$ and number of reductions is 3. So, $n - 1$

👍 73 votes -- (points)

2.15.34 Parsing: GATE CSE 2014 Set 1 | Question: 34 top https://gateoverflow.in/1807

✔ The question is asked with respect to the symbol $'<'$ which is **not present** in the given canonical set of items. Hence it is neither a shift-reduce conflict nor a reduce-reduce conflict on symbol '<'. Hence **D** is the correct option. But if the question would have asked with respect to the symbol $'>'$ then it would have been a shift-reduce conflict.

👍 31 votes -- Divya Bharti (8.8k points)

2.15.35 Parsing: GATE CSE 2015 Set 3 | Question: 16 top https://gateoverflow.in/8413

✔ Answer is **C**.

SLR is the simplest to implement and Canonical LR is the most powerful.

http://en.wikipedia.org/wiki/LALR_parser_generator

References

👍 39 votes -- Arjun Suresh (330k points)

2.15.36 Parsing: GATE CSE 2015 Set 3 | Question: 31 top https://gateoverflow.in/8488

✔ **A parser works on the basis of given grammar. It takes the grammar as it is. Parser does not work on the**

basis of the yield of the grammar. Also, while constructing the LL(1) parser table, that entry for terminal 'c' will contain multiple entries. So, LL(1) parser cannot be constructed for the given grammar.

$S \to F|H$

$F \to p|c$

$H \to d|c$

That $\{p, d, c\}$ are the strings generated by the grammar is absolutely correct. But LL(1) and LR(1) can parse these strings successfully only if the grammar is unambiguous and like given below...

$S \to P|D|C$

$P \to p$

$D \to d$

$C \to c$

Please note the difference between these two grammars. Both derive the same strings, but in different manner. With the grammar given in the question, both top-down and bottom-up parsers will get confused while deriving "c". Top-down parser will get confused between $F \to c$ and $H \to c$. Similarly, bottom-up parser will get confused while reducing " c". This confusion in case of bottom-up parsing is technically termed as "reduce-reduce" conflict.

While top-down parsing, first(F) and first(H) are not disjoint, so the grammar cannot be LL(1). Therefore, LL(1) parser cannot parse it.

Hence, the answer should be option (**D**). Neither S1 nor S2.

54 votes -- ashishacm (263 points)

2.15.37 Parsing: GATE CSE 2016 Set 1 | Question: 45 top https://gateoverflow.in/39697

✔ $2 - 5 + 1 - 7 * 3$ will be evaluated according to the precedence and associativity as given in the question as follows:
$((2 - ((5 + 1) - 7)) * 3) \Rightarrow ((2 - (-1)) * 3) \Rightarrow 9$

73 votes -- Monanshi Jain (7k points)

2.15.38 Parsing: GATE CSE 2017 Set 1 | Question: 17 top https://gateoverflow.in/118297

✔ Follow of Q is first of R so we get $\{w\}$

but since R can be Null so we have to check first of S which is $\{y\}$

so FOLLOW $Q = \{w, y\}$

Correct option (**C**)

20 votes -- sriv_shubham (2.8k points)

2.15.39 Parsing: GATE CSE 2017 Set 1 | Question: 43 top https://gateoverflow.in/118326

✔ This question is picked from area of **Counting in Combinatorics**.

Given:
if e_1 then e_2 else e_3 has 2 control flow paths $e_1 \to e_2$ and $e_1 \to e_3$.
(Meaning of **"how many control flow"** for if structure is clearly mentioned)

What is asked:
Number of control flow paths for 10 if terminals?

Solution:
To get 10 if's we need to use grammar to get,
if <expr> then <expr> else <expr> ; stmt
if <expr> then <expr> else <expr> ; if <expr> then <expr> else <expr> ; stmt
....
....
....
(keep doing it 10 times to get 10 if's)
Observe that there is a **semi-colon after every if structure**.

We know that every if structure has 2 control flows as given in question. Hence,
We have 2 control flow choices for 1st if terminal.
We have 2 control flow choices for 2nd if terminal.
....
....
....
We have 2 control flow choices for 10th if terminal.

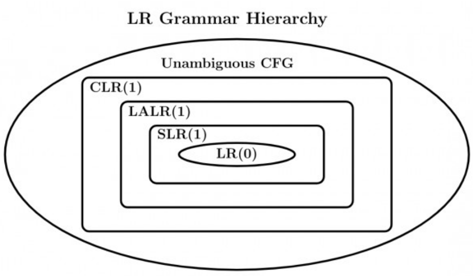
By using multiplicative law of counting we get,
Total choices as $2 * 2 * 2 * 2 * 2......10$ times $= 2^{10} = 1024$

Once again, **one need not know "what control flow"** is, but needs to know **"how many control flows" are in if structure** which is given in question.

👍 67 votes -- prs11 (213 points)

2.15.40 Parsing: GATE CSE 2017 Set 2 | Question: 6 top https://gateoverflow.in/118343

✔



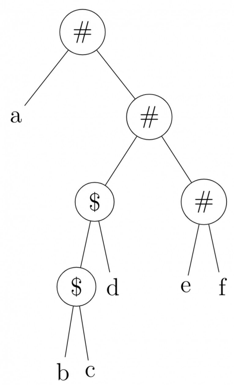
For a parser more power means it can parse more strings. So, here only the first statement is correct.

Correct Answer: A

👍 25 votes -- 2018 (5.5k points)

2.15.41 Parsing: GATE CSE 2018 | Question: 38 top https://gateoverflow.in/204112

✔



Inorder : $\{a\#[((b\$c)\$d)\#(e\#f)]\}$ (given in question)

If we observe, first evaluation is $b\$c$
So, ($) has higher priority.
Therefore, either option (A) or (C) is correct

Option A

$ has higher precedence and $\#$ is right associative.

From tree, it is clear that $(e\#f)$ is evaluating first which is to the right side of the root.
Therefore, $\#$ is Right Associative.
So, Option A is correct

Option C

$ has higher precedence and $\#$ is left associative.
This is wrong.

Correct Answer: A.

20 votes -- venu7850 (131 points)

2.15.42 Parsing: GATE CSE 2019 | Question: 19 top https://gateoverflow.in/302829

✔ For Follow(B) $\implies$ First(D) $= \{d, \epsilon\}$

Put ϵ in II production

Follow (B) $=$ Follow (A) $= \{a\}$

Follow (B) $= \{d, a\}$

According to the question writing Follow set in decreasing order:

a	d
3	1

Hence 31 is correct answer

21 votes -- Ashwani Kumar (13k points)

2.15.43 Parsing: GATE CSE 2019 | Question: 3 top https://gateoverflow.in/302845

✔ A bottom-up parser traces a rightmost derivation in reverse. **Answer (D)**.

25 votes -- Digvijay (44.9k points)

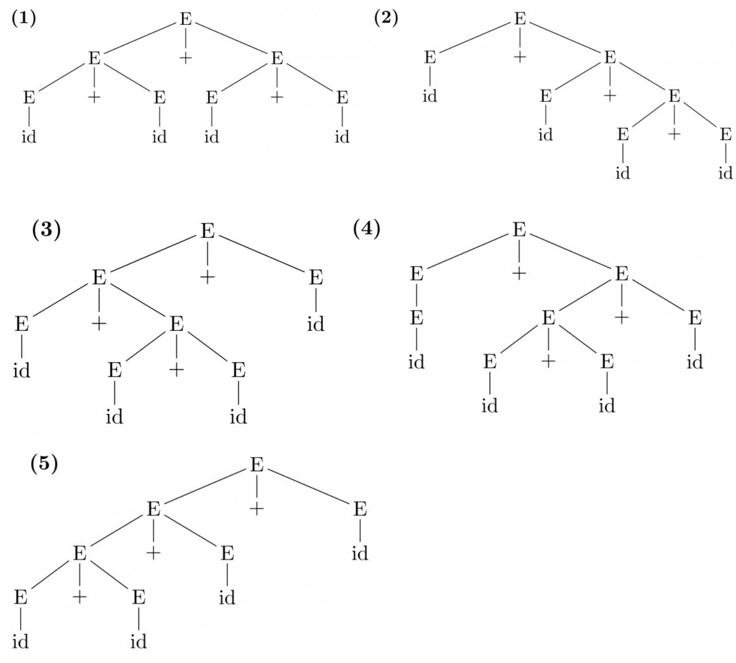
2.15.44 Parsing: GATE IT 2005 | Question: 83a top https://gateoverflow.in/3849

✔ Answer is **A**.

25 votes -- Gate Keeda (15.9k points)

2.15.45 Parsing: GATE IT 2005 | Question: 83b top https://gateoverflow.in/3850

✔ 5 Parse trees are possible



25 votes -- Shaji Thorn Blue (5.1k points)

2.15.46 Parsing: GATE IT 2008 | Question: 79 top https://gateoverflow.in/3393

✔ $S \underset{1}{\rightarrow} aS$

$\underset{2}{\rightarrow} aA$
$\underset{3}{\rightarrow} aaAb$
$\underset{4}{\rightarrow} aabAab$
$\underset{5}{\rightarrow} aabbAaab$
$\underset{6}{\rightarrow} aabbaab$

Thus 6 steps are needed and only one way to derive the string so only one parse tree.

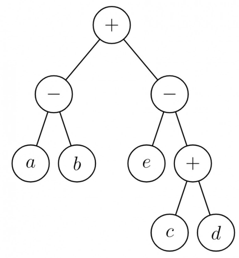
Correct Answer: A

43 votes -- Shreyans Dhankhar (2.1k points)

2.16 Register Allocation (2) top

2.16.1 Register Allocation: GATE CSE 2011 | Question: 36 top https://gateoverflow.in/2138

Consider evaluating the following expression tree on a machine with load-store architecture in which memory can be accessed only through load and store instructions. The variables $a, b, c, d,$ and e are initially stored in memory. The binary operators used in this expression tree can be evaluated by the machine only when operands are in registers. The instructions produce result only in a register. If no intermediate results can be stored in memory, what is the minimum number of registers needed to evaluate this expression?



A. 2
B. 9
C. 5
D. 3

gate2011-cse compiler-design register-allocation normal

Answer

2.16.2 Register Allocation: GATE CSE 2017 Set 1 | Question: 52 top https://gateoverflow.in/118746

Consider the expression $(a - 1) * (((b + c)/3) + d)$. Let X be the minimum number of registers required by an *optimal* code generation (without any register spill) algorithm for a load/store architecture, in which

A. only load and store instructions can have memory operands and
B. arithmetic instructions can have only register or immediate operands.

The value of X is _____________ .

gate2017-cse-set1 compiler-design register-allocation normal numerical-answers

Answer

Answers: Register Allocation

2.16.1 Register Allocation: GATE CSE 2011 | Question: 36 top https://gateoverflow.in/2138

✔ Given is Load Store Architecture, that means we can access memory using Load and Store Instructions.

Key Idea:- Pick new register only when it is required.

We want to add **c** and **d**, and initially both are in memory, therefore copy these into registers.

- load $R1, c \quad (R1 \leftarrow c)$
- load $R2, d \quad (R2 \leftarrow d)$

(here no compensation can be done, we need two registers)

- add $R1, R1, R2$ $(R1 \leftarrow R1 + R2)$

(at this point $R1$ is holding $\mathbf{c+d}$ and $R2$ is holding $\mathbf{d}$, i.e. $R1 \leftarrow \mathbf{c+d}$ and $R2 \leftarrow \mathbf{d}$)
Now, $\mathbf{e}$ comes into picture and my question is, Can i make use of $R1$ or $R2$ to store $\mathbf{e}$?
I can not use R1 to store **e** as its value will be needed later but I can use R2.

- load $R2, e$

(currently $R1 \leftarrow \mathbf{c+d}$ and $R2 \leftarrow \mathbf{e}$)

- Sub $R1, R2, R1$ $(R1 \leftarrow R2 - R1)$

Doing this all gives, final value of right sub-tree is stored in $R1$, and $R2$ stores $\mathbf{e}$.
Now, coming to left subtree, to perform "$a - b$" we need to copy both variables in registers.
We can copy one of the variable in $R2$, but we can not obviously copy in $R1$ as value of $R1$ will be required later.

- Load $R2, a$
- Load $R3, b$ (**here comes extra register, and we can not avoid using it** .)

Current mapping is $R2 \leftarrow a$, $R3 \leftarrow b$ and $R1$ contains final value of Right subtree.

- SUB $R2, R2, R3 (R2 \leftarrow R2 - R3)$
- ADD $R1, R1, R2$

Hence answer is 3 i.e. **D**

70 votes -- Sachin Mittal (15.8k points)

2.16.2 Register Allocation: GATE CSE 2017 Set 1 | Question: 52 top https://gateoverflow.in/118746

✔ Load $R1, b$

Load $R2, c$

ADD $R1, R2$

Div $R1, 3$

Load $R2, d$

Add $R1, R2$

Load $R2, a$

Sub $R2, 1$

Mul $R2, R1$

hence minimum 2 registers required

73 votes -- sriv_shubham (2.8k points)

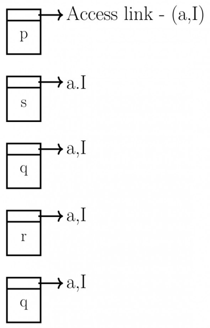
2.17 Runtime Environments (19) top

2.17.1 Runtime Environments: GATE CSE 1988 | Question: 2xii top https://gateoverflow.in/93966

Consider the following program skeleton and below figure which shows activation records of procedures involved in the calling sequence.

$$p \rightarrow s \rightarrow q \rightarrow r \rightarrow q.$$

Write the access links of the activation records to enable correct access and variables in the procedures from other procedures involved in the calling sequence




```
procedure p;
  procedure q;
    procedure r;
      begin
        q
      end r;
    begin
      r
    end q;
  procedure s;
    begin
      q
    end s;
  begin
    s
  end p;
```

gate1988 normal descriptive runtime-environments compiler-design

Answer

2.17.2 Runtime Environments: GATE CSE 1989 | Question: 10a top

https://gateoverflow.in/89636

Will recursion work correctly in a language with static allocation of all variables? Explain.

gate1989 descriptive compiler-design runtime-environments

Answer

2.17.3 Runtime Environments: GATE CSE 1989 | Question: 8b top

https://gateoverflow.in/89082

Indicate the result of the following program if the language uses (i) static scope rules and (ii) dynamic scope rules.

```
        var x, y:integer;
        procedure A (var z:integer);
        var x:integer;
        begin x:=1; B; z:= x end;
        procedure B;
        begin x:=x+1 end;
        begin
        x:=5; A(y); write (y)
...end.
```

gate1989 descriptive compiler-design runtime-environments

Answer

2.17.4 Runtime Environments: GATE CSE 1990 | Question: 2-v top

https://gateoverflow.in/83980

Match the pairs in the following questions:

(a)	Pointer data type	(p)	Type conversion
(b)	Activation record	(q)	Dynamic data structure
(c)	Repeat-until	(r)	Recursion
(d)	Coercion	(s)	Nondeterministic loop

gate1990 match-the-following compiler-design runtime-environments recursion

Answer

2.17.5 Runtime Environments: GATE CSE 1990 | Question: 4-v top

https://gateoverflow.in/85394

State whether the following statements are TRUE or FALSE with reason:

The Link-load-and-go loading scheme required less storage space than the link-and-go loading scheme.

gate1990 true-false compiler-design runtime-environments

Answer

2.17.6 Runtime Environments: GATE CSE 1993 | Question: 7.7 top

https://gateoverflow.in/2295

A part of the system software which under all circumstances must reside in the main memory is:

A. text editor
B. assembler
C. linker
D. loader
E. none of the above

gate1993 compiler-design runtime-environments easy

Answer

2.17.7 Runtime Environments: GATE CSE 1995 | Question: 1.14 top

https://gateoverflow.in/2601

A linker is given object modules for a set of programs that were compiled separately. What information need to be included in an object module?

A. Object code
B. Relocation bits
C. Names and locations of all external symbols defined in the object module
D. Absolute addresses of internal symbols

gate1995 compiler-design runtime-environments normal

Answer

2.17.8 Runtime Environments: GATE CSE 1997 | Question: 1.10 top

https://gateoverflow.in/2226

Heap allocation is required for languages.

A. that support recursion
B. that support dynamic data structure
C. that use dynamic scope rules
D. None of the above

gate1997 compiler-design easy runtime-environments

Answer

2.17.9 Runtime Environments: GATE CSE 1997 | Question: 1.8 top

https://gateoverflow.in/2224

A language L allows declaration of arrays whose sizes are not known during compilation. It is required to make efficient use of memory. Which one of the following is true?

A. A compiler using static memory allocation can be written for L
B. A compiler cannot be written for L; an interpreter must be used
C. A compiler using dynamic memory allocation can be written for L
D. None of the above

gate1997 compiler-design easy runtime-environments

Answer

2.17.10 Runtime Environments: GATE CSE 1998 | Question: 1.25, ISRO2008-41 top https://gateoverflow.in/1662

In a resident – OS computer, which of the following systems must reside in the main memory under all situations?

A. Assembler
B. Linker
C. Loader
D. Compiler

gate1998 compiler-design runtime-environments normal isro2008

Answer

2.17.11 Runtime Environments: GATE CSE 1998 | Question: 1.28 top https://gateoverflow.in/1665

A linker reads four modules whose lengths are $200, 800, 600$ and 500 words, respectively. If they are loaded in that order, what are the relocation constants?

A. $0, 200, 500, 600$
B. $0, 200, 1000, 1600$
C. $200, 500, 600, 800$
D. $200, 700, 1300, 2100$

gate1998 compiler-design runtime-environments normal

Answer

2.17.12 Runtime Environments: GATE CSE 1998 | Question: 2.15 top https://gateoverflow.in/1687

Faster access to non-local variables is achieved using an array of pointers to activation records called a

A. stack
B. heap
C. display
D. activation tree

gate1998 programming compiler-design normal runtime-environments

Answer

2.17.13 Runtime Environments: GATE CSE 2001 | Question: 1.17 top https://gateoverflow.in/710

The process of assigning load addresses to the various parts of the program and adjusting the code and the data in the program to reflect the assigned addresses is called

A. Assembly
B. parsing
C. Relocation
D. Symbol resolution

gate2001-cse compiler-design runtime-environments easy

Answer

2.17.14 Runtime Environments: GATE CSE 2008 | Question: 54 top https://gateoverflow.in/477

Which of the following are true?

I. A programming language which does not permit global variables of any kind and has no nesting of procedures/functions, but permits recursion can be implemented with static storage allocation
II. Multi-level access link (or display) arrangement is needed to arrange activation records only if the programming language being implemented has nesting of procedures/functions
III. Recursion in programming languages cannot be implemented with dynamic storage allocation

IV. Nesting procedures/functions and recursion require a dynamic heap allocation scheme and cannot be implemented with a stack-based allocation scheme for activation records
V. Programming languages which permit a function to return a function as its result cannot be implemented with a stack-based storage allocation scheme for activation records

A. II and V only
B. I, III and IV only
C. I, II and V only
D. II, III and V only

gate2008-cse compiler-design difficult runtime-environments

Answer

2.17.15 Runtime Environments: GATE CSE 2010 | Question: 14 top

https://gateoverflow.in/2187

Which languages necessarily need heap allocation in the runtime environment?

A. Those that support recursion.
B. Those that use dynamic scoping.
C. Those that allow dynamic data structure.
D. Those that use global variables.

gate2010-cse compiler-design easy runtime-environments

Answer

2.17.16 Runtime Environments: GATE CSE 2012 | Question: 36 top

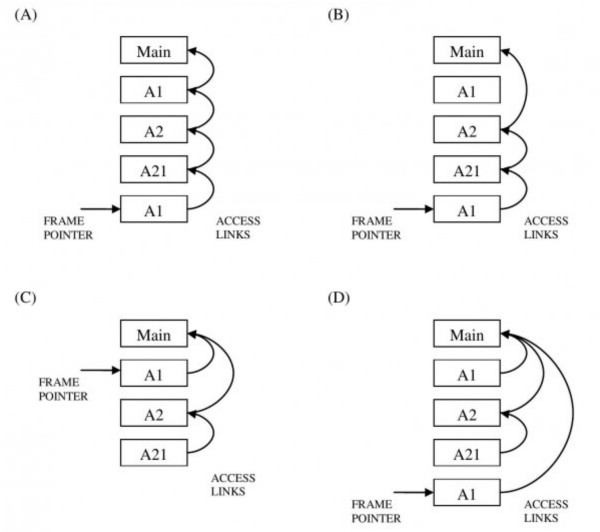
https://gateoverflow.in/1758

Consider the program given below, in a block-structured pseudo-language with lexical scoping and nesting of procedures permitted.

```
Program main;
  Var ...

  Procedure A1;
    Var ...
    Call A2;
  End A1

  Procedure A2;
    Var ...

    Procedure A21;
      Var ...
      Call A1;
    End A21

    Call A21;
  End A2

  Call A1;
End main.
```

Consider the calling chain: $Main \rightarrow A1 \rightarrow A2 \rightarrow A21 \rightarrow A1$

The correct set of activation records along with their access links is given by:



gate2012-cse compiler-design runtime-environments normal

Answer

2.17.17 Runtime Environments: GATE CSE 2014 Set 2 | Question: 18 top

https://gateoverflow.in/1975

Which one of the following is **NOT** performed during compilation?

A. Dynamic memory allocation
B. Type checking
C. Symbol table management
D. Inline expansion

gate2014-cse-set2 compiler-design easy runtime-environments

Answer

2.17.18 Runtime Environments: GATE CSE 2014 Set 3 | Question: 18 top

https://gateoverflow.in/2052

Which of the following statements are CORRECT?

1. Static allocation of all data areas by a compiler makes it impossible to implement recursion.
2. Automatic garbage collection is essential to implement recursion.
3. Dynamic allocation of activation records is essential to implement recursion.
4. Both heap and stack are essential to implement recursion.

A. 1 and 2 only
B. 2 and 3 only
C. 3 and 4 only
D. 1 and 3 only

gate2014-cse-set3 compiler-design runtime-environments normal

Answer

2.17.19 Runtime Environments: GATE CSE 2021 Set 1 | Question: 4 top

https://gateoverflow.in/357448

Consider the following statements.

- S_1 : The sequence of procedure calls corresponds to a preorder traversal of the activation tree.
- S_2 : The sequence of procedure returns corresponds to a postorder traversal of the activation tree.

Which one of the following options is correct?

A. S_1 is true and S_2 is false
B. S_1 is false and S_2 is true
C. S_1 is true and S_2 is true
D. S_1 is false and S_2 is false

gate2021-cse-set1 runtime-environments normal

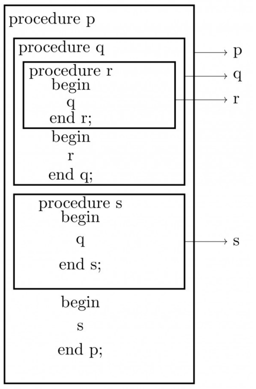
Answer

Answers: Runtime Environments

2.17.1 Runtime Environments: GATE CSE 1988 | Question: 2xii top

☛ https://gateoverflow.in/93966

✔



When procedure p begins, s is called $[p \to s]$. s is enclosed within p. So, any undeclared variable found inside s will be searched for inside the body within which it is enclosed i.e., function p (static scoping rules). **So access link is from s to p**.

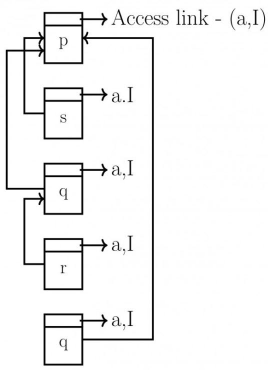
Now, s is visited and s calls q $[p \to s \to q]$. q is visited and begin execution. Any undeclared variable found inside q will be searched inside the enclosing function i.e. p again. **So, access link is from q to p**.

Now, q calls r $[p \to s \to q \to r]$. r is visited. r begins execution and any undeclared variable found is searched in the enclosing function i.e., q here. **So, access link is from r to q**.

r calls q $[p \to s \to q \to r \to q]$: This is the sequence of calls which is already given in the question.

We have already seen access links for q.

So, it becomes



👍 29 votes -- Sudeshna Chaudhuri (17.4k points)

2.17.2 Runtime Environments: GATE CSE 1989 | Question: 10a top

☛ https://gateoverflow.in/89636

Recursion can not be used for static variables since the memory for static variables are allocated only once . so if we use recursion, then it may give incorrect result . Generally, we do not want such things when implementing recursion.

For detailed explanation, refer http://stackoverflow.com/questions/10290527/why-do-static-variables-not-allow-recursion

References

👍 13 votes -- Sonu Kumar (387 points)

2.17.3 Runtime Environments: GATE CSE 1989 | Question: 8b top

https://gateoverflow.in/89082

Using static rules, we get 1.

Using dynamic rules, we get 2

2 votes -- kapilbk1996 (387 points)

2.17.4 Runtime Environments: GATE CSE 1990 | Question: 2-v top

https://gateoverflow.in/83980

✔

(a)	Pointer data type	(q)	Dynamic data structure
(b)	Activation record	(r)	Recursion
(c)	Repeat-until	(s)	Nondeterministic loop
(d)	Coercion	(p)	Type conversion

23 votes -- Digvijay (44.9k points)

2.17.5 Runtime Environments: GATE CSE 1990 | Question: 4-v top

https://gateoverflow.in/85394

TRUE.

In link and go scheme the linkage editor coexists with program in main memory while performing linking task whereas link,load and go scheme the linkage editor does not coexists with program in main memory while performing linking task

source : http://www.answers.com/Q/What_are_link_and_go_and_link_load_and_go_loader_schemes

References

18 votes -- junk_mayavi (3k points)

2.17.6 Runtime Environments: GATE CSE 1993 | Question: 7.7 top

https://gateoverflow.in/2295

✔ Answer: **D**

The loader is a program that loads the object program from the secondary memory into the main memory for execution of the program. The loader resides in main memory.

23 votes -- Rajarshi Sarkar (27.8k points)

2.17.7 Runtime Environments: GATE CSE 1995 | Question: 1.14 top

https://gateoverflow.in/2601

✔

A. is trivially there is an object module.

B. must be there if we need to have relocation capability.

C. is the **answer**. For linker to link external symbols (for example in C, to link an extern variable in one module to a global variable in another module), it must know the location of all external symbols. In C external symbols includes all global variables and function names.

D. is no way needed.

53 votes -- Arjun Suresh (330k points)

2.17.8 Runtime Environments: GATE CSE 1997 | Question: 1.10 top

https://gateoverflow.in/2226

✔ Memory is taken from heap for dynamic allocation.

So, option (**B**) is correct.

20 votes -- Sankaranarayanan P.N (8.5k points)

2.17.9 Runtime Environments: GATE CSE 1997 | Question: 1.8 top https://gateoverflow.in/2224

✔ **C**.

Using dynamic memory allocation, memory will be allocated to array at runtime.

24 votes -- Gate Keeda (15.9k points)

2.17.10 Runtime Environments: GATE CSE 1998 | Question: 1.25, ISRO2008-41 top https://gateoverflow.in/1662

✔ **(C)** is answer.
In many operating systems the loader is permanently resident in memory, although some operating systems that support virtual memory may allow the loader to be located in a region of memory that is pageable.

Reference: Loader

References

32 votes -- Mithlesh Upadhyay (4.3k points)

2.17.11 Runtime Environments: GATE CSE 1998 | Question: 1.28 top https://gateoverflow.in/1665

✔ answer - **B**

first module loaded starting at address 0. Size is 200. hence it will occup first 200 address last address being 199. Second module will be present from 200 and so on.

25 votes -- Ankit Rokde (6.9k points)

2.17.12 Runtime Environments: GATE CSE 1998 | Question: 2.15 top https://gateoverflow.in/1687

✔ Correct Option: ***C***

Properties of displays

1. Use a pointer array to store the activation records along the static chain.
2. Fast access for non-local but may be complicated to maintain.
3. Calling a subprogram in the same level – simply replace and restore.
4. Calling a subprogram in the higher level – add an entry and may need to save the old pointers.
5. Calling a subprogram in the lower level – shrink the pointer and restore it when the subprogram returns.

http://users.dickinson.edu/~wahlst/356/ch10.pdf

References

38 votes -- sumit kumar singh dixit (1.6k points)

2.17.13 Runtime Environments: GATE CSE 2001 | Question: 1.17 top https://gateoverflow.in/710

✔ **Relocation** is the process of assigning load addresses to position-dependent code of a program and adjusting the code and data in the program to reflect the assigned addresses.

Hence Option C is Ans

Symbol resolution is the process of searching files and libraries to replace symbolic references or names of libraries with actual usable addresses in memory before running a program.

45 votes -- Rajesh Pradhan (18.9k points)

2.17.14 Runtime Environments: GATE CSE 2008 | Question: 54 top

☛ https://gateoverflow.in/477

✔

I. False. Recursion cannot be implemented using static allocation.
II. True. Yes, we do need multi level access link in case of nested functions. Each level to traverse ARB of same level of nesting.
III. False. Recursion can only be implemented using dynamic memory allocation.
IV. False. Recursion is done using memory in stack (ARBs in stack), not in heap.
V. True. Yes, they cannot, once a function returns its activation record is no longer valid, so we cannot return a function as a result.

So, option (**A**) is correct.

26 votes -- Monanshi Jain (7k points)

2.17.15 Runtime Environments: GATE CSE 2010 | Question: 14 top

☛ https://gateoverflow.in/2187

✔ Those that allow dynamic data structure.

malloc etc uses memory from heap area.

32 votes -- Sankaranarayanan P.N (8.5k points)

2.17.16 Runtime Environments: GATE CSE 2012 | Question: 36 top

☛ https://gateoverflow.in/1758

✔ **Activation record:**

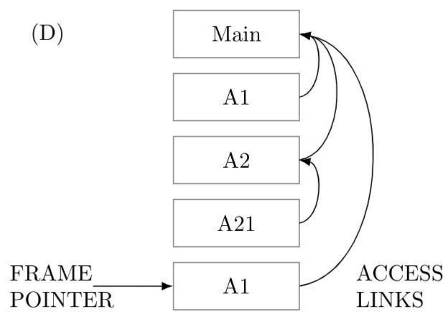
Temporaries
local data
machine status
Access links
Control links
Parameters
Return value

Access link: to access non local data

The calling chain: Main→A1→A2→A21→A1

PROCEDURE	non-local data present
A1	outside procedure A1 body i.e. in main procedure So, A1---> main
A2	outside procedure A2 body i.e. in main procedure So, A2---> main
A21	outside procedure A21 body i.e. in A2 procedure So, A21---> A2

(D)



https://www.youtube.com/watch?v=mMK-TlvH5c4&t=1093s @18:00

References

12 votes -- ANIKET KUMAR (597 points)

2.17.17 Runtime Environments: GATE CSE 2014 Set 2 | Question: 18 top https://gateoverflow.in/1975

✔ Dynamic means- at runtime. Dynamic memory allocation happens during the execution time and hence (**A**) is the answer.

38 votes -- Arjun Suresh (330k points)

2.17.18 Runtime Environments: GATE CSE 2014 Set 3 | Question: 18 top https://gateoverflow.in/2052

✔ It will be **D**.

option 2 is wrong because it is not necessary to have automatic garbage collection to implement recursion.

option 4 is wrong because it says that both are required to implement recursion, which is wrong. Either of them will suffice.

37 votes -- Gate Keeda (15.9k points)

2.17.19 Runtime Environments: GATE CSE 2021 Set 1 | Question: 4 top https://gateoverflow.in/357448

✔ **Correct Answer: C**

S_1: Is true because to perform procedure calls, first parent function will call child functions and hence it resembles **preorder**.

S_2: Is true because to return parent function , we must return child functions first. Hence it resembles **post order.**

4 votes -- Sherrinford03 (635 points)

2.18 Static Single Assignment (2) top

2.18.1 Static Single Assignment: GATE CSE 2016 Set 1 | Question: 19 top https://gateoverflow.in/39675

Consider the following code segment.

```
x = u - t;
y = x * v;
x = y + w;
y = t - z;
y = x * y;
```

The minimum number of *total* variables required to convert the above code segment to *static single assignment* form is ________.

gate2016-cse-set1 compiler-design static-single-assignment normal numerical-answers

Answer

2.18.2 Static Single Assignment: GATE CSE 2017 Set 1 | Question: 12 top

☛ https://gateoverflow.in/118292

Consider the following intermediate program in three address code

```
p = a - b
q = p * c
p = u * v
q = p + q
```

Which one of the following corresponds to a *static single assignment* form of the above code?

A.
```
p1 = a  - b
q1 = p1 * c
p1 = u  * v
q1 = p1 + q1
```

B.
```
p3 = a  - b
q4 = p3 * c
p4 = u  * v
q5 = p4 + q4
```

C.
```
p1 = a  - b
q1 = p2 * c
p3 = u  * v
q2 = p4 + q3
```

D.
```
p1 = a - b
q1 = p * c
p2 = u * v
q2 = p + q
```

gate2017-cse-set1 compiler-design intermediate-code normal static-single-assignment

Answer

Answers: Static Single Assignment

2.18.1 Static Single Assignment: GATE CSE 2016 Set 1 | Question: 19 top

☛ https://gateoverflow.in/39675

✔ In Static Single Assignment when we assign the values, the variables to which the value is being assigned should be unique.

$T1 = u - t$

$T2 = T1 * v$

$T3 = T2 + w$

$T4 = t - z$

$T5 = t3 * t4$

So $T1.....T5 = 5 + (u, t, v, w, z) = 5$

Total **10** variables.

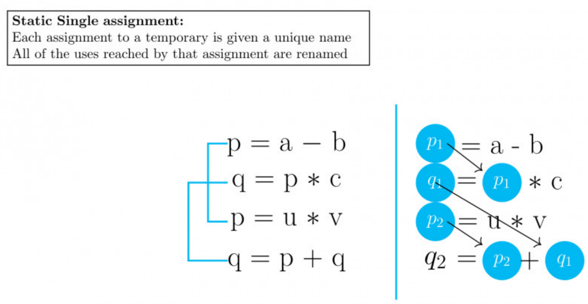
Note: RHS of the operation can use the previously used variables, but LHS in SSA must always be unique.

79 votes -- Prateek Khade (2.1k points)

2.18.2 Static Single Assignment: GATE CSE 2017 Set 1 | Question: 12 top

☛ https://gateoverflow.in/118292

✔



https://en.wikipedia.org/wiki/Static_single_assignment_form http://www.cse.iitd.ernet.in/~nvkrishna/courses/winter07/ssa.pdf

https://www.cs.cmu.edu/~fp/courses/15411-f08/lectures/09-ssa.pdf

http://www.seas.harvard.edu/courses/cs252/2011sp/slides/Lec04-SSA.pdf

So, **B** is ans.

References

36 votes -- 2018 (5.5k points)

2.19 Syntax Directed Translation (11) top

2.19.1 Syntax Directed Translation: GATE CSE 1992 | Question: 11a top https://gateoverflow.in/590

Write syntax directed definitions (semantic rules) for the following grammar to add the type of each identifier to its entry in the symbol table during semantic analysis. Rewriting the grammar is not permitted and semantic rules are to be added to the ends of productions only.

- $D \to TL;$
- $T \to \text{int}$
- $T \to \text{real}$
- $L \to L, id$
- $L \to id$

gate1992 compiler-design syntax-directed-translation normal descriptive

Answer

2.19.2 Syntax Directed Translation: GATE CSE 1995 | Question: 2.10 top https://gateoverflow.in/2622

A shift reduce parser carries out the actions specified within braces immediately after reducing with the corresponding rule of grammar

- $S \to xxW\ \{\text{print}\,\text{"1"}\}$
- $S \to y\ \{\text{print}\,\text{"2"}\}$
- $W \to Sz\ \{\text{print}\,\text{"3"}\}$

What is the translation of $xxxxyzz$ using the syntax directed translation scheme described by the above rules?

A. 23131
B. 11233
C. 11231
D. 33211

gate1995 compiler-design grammar syntax-directed-translation normal

Answer

2.19.3 Syntax Directed Translation: GATE CSE 1996 | Question: 20 top https://gateoverflow.in/2772

Consider the syntax-directed translation schema (SDTS) shown below:

- $E \to E + E$ {print "+"}
- $E \to E * E$ {print "."}
- $E \to id$ {print id.name}
- $E \to (E)$

An LR-parser executes the actions associated with the productions immediately after a reduction by the corresponding production. Draw the parse tree and write the translation for the sentence.

$(a + b) * (c + d)$, using SDTS given above.

gate1996 compiler-design syntax-directed-translation normal descriptive

Answer

2.19.4 Syntax Directed Translation: GATE CSE 1998 | Question: 23 top https://gateoverflow.in/1738

Let the attribute 'val' give the value of a binary number generated by S in the following grammar:

- $S \to L.L \mid L$
- $L \to LB \mid B$
- $B \to 0 \mid 1$

For example, an input 101.101 gives $S.val = 5.625$

Construct a syntax directed translation scheme using only synthesized attributes, to determine $S.val$.

gate1998 compiler-design syntax-directed-translation normal descriptive

Answer

2.19.5 Syntax Directed Translation: GATE CSE 2000 | Question: 19 top https://gateoverflow.in/690

Consider the syntax directed translation scheme (SDTS) given in the following. Assume attribute evaluation with bottom-up parsing, i.e., attributes are evaluated immediately after a reduction.

- $E \to E_1 * T \quad \{E.val = E_1.val * T.val\}$
- $E \to T \quad \{E.val = T.val\}$
- $T \to F - T_1 \quad \{T.val = F.val - T_1.val\}$
- $T \to F \quad \{T.val = F.val\}$
- $F \to 2 \quad \{F.val = 2\}$
- $F \to 4 \quad \{F.val = 4\}$

A. Using this SDTS, construct a parse tree for the expression $4 - 2 - 4 * 2$ and also compute its $E.val$.
B. It is required to compute the total number of reductions performed to parse a given input. Using synthesized attributes only, modify the SDTS given, without changing the grammar, to find $E.red$, the number of reductions performed while reducing an input to E.

gate2000-cse compiler-design syntax-directed-translation normal descriptive

Answer

2.19.6 Syntax Directed Translation: GATE CSE 2001 | Question: 17 top https://gateoverflow.in/758

The syntax of the repeat-until statement is given by the following grammar

$S \to$ repeat S_1 until E

where E stands for expressions, S and S_1 stand for statements. The non-terminals S and S_1 have an attribute code that represents generated code. The non-terminal E has two attributes. The attribute code represents generated code to evaluate the expression and store its value in a distinct variable, and the attribute varName contains the name of the variable in which the truth value is stored. The truth-value stored in the variable is 1 if E is true, 0 if E is false.

Give a syntax-directed definition to generate three-address code for the repeat-until statement. Assume that you can call a function newlabel() that returns a distinct label for a statement. Use the operator '\\' to concatenate two strings and the function gen(s) to generate a line containing the string s.

gate2001-cse compiler-design syntax-directed-translation normal descriptive

Answer

2.19.7 Syntax Directed Translation: GATE CSE 2003 | Question: 18 top https://gateoverflow.in/908

In a bottom-up evaluation of a syntax directed definition, inherited attributes can

A. always be evaluated
B. be evaluated only if the definition is L-attributed
C. be evaluated only if the definition has synthesized attributes
D. never be evaluated

gate2003-cse compiler-design syntax-directed-translation normal

Answer

2.19.8 Syntax Directed Translation: GATE CSE 2016 Set 1 | Question: 46 top https://gateoverflow.in/39700

Consider the following Syntax Directed Translation Scheme $(SDTS)$, with non-terminals $\{S, A\}$ and terminals $\{a, b\}$.

$$S \rightarrow aA \quad \{\text{print } 1\}$$
$$S \rightarrow a \quad \{\text{print } 2\}$$
$$A \rightarrow Sb \quad \{\text{print } 3\}$$

Using the above $SDTS$, the output printed by a bottom-up parser, for the input aab is:

A. 1 3 2
B. 2 2 3
C. 2 3 1
D. syntax error

gate2016-cse-set1 compiler-design syntax-directed-translation normal

Answer

2.19.9 Syntax Directed Translation: GATE CSE 2019 | Question: 36 top https://gateoverflow.in/302812

Consider the following grammar and the semantic actions to support the inherited type declaration attributes. Let X_1, X_2, X_3, X_4, X_5 , and X_6 be the placeholders for the non-terminals D, T, L or L_1 in the following table:

Production rule	Semantic action
$D \rightarrow TL$	$X_1.\text{type} = X_2.\text{type}$
$T \rightarrow \text{int}$	$T.\text{type} = \text{int}$
$T \rightarrow \text{float}$	$T.\text{type} = \text{float}$
$L \rightarrow L_1, id$	$X_3.\text{type} = X_4.\text{type}$ $\text{addType}(id.\text{entry}, X_5.\text{type})$
$L \rightarrow id$	$\text{addType}(id.\text{entry}, X_6.\text{type})$

Which one of the following are appropriate choices for X_1, X_2, X_3 and X_4?

A. $X_1 = L,\ X_2 = T,\ X_3 = L_1,\ X_4 = L$
B. $X_1 = T,\ X_2 = L,\ X_3 = L_1,\ X_4 = T$
C. $X_1 = L,\ X_2 = L,\ X_3 = L_1,\ X_4 = T$
D. $X_1 = T,\ X_2 = L,\ X_3 = T,\ X_4 = L_1$

gate2019-cse compiler-design syntax-directed-translation

Answer

2.19.10 Syntax Directed Translation: GATE CSE 2020 | Question: 33 top https://gateoverflow.in/333198

Consider the productions $A \rightarrow PQ$ and $A \rightarrow XY$. Each of the five non-terminals $A, P, Q, X,$ and Y has two attributes: s is a synthesized attribute, and i is an inherited attribute. Consider the following rules.

- Rule 1 : $P.i = A.i + 2,\ Q.i = P.i + A.i,$ and $A.s = P.s + Q.s$
- Rule 2 : $X.i = A.i + Y.s$ and $Y.i = X.s + A.i$

Which one of the following is TRUE?

A. Both Rule 1 and Rule 2 are L-attributed.
B. Only Rule 1 is L-attributed.
C. Only Rule 2 is L-attributed.
D. Neither Rule 1 nor Rule 2 is L-attributed.

gate2020-cse compiler-design syntax-directed-translation

Answer

2.19.11 Syntax Directed Translation: GATE CSE 2021 Set 1 | Question: 26 top

https://gateoverflow.in/357425

Consider the following grammar (that admits a series of declarations, followed by expressions) and the associated syntax directed translation (SDT) actions, given as pseudo-code

$$
\begin{array}{lll}
P & \rightarrow & D^* E^* \\
D & \rightarrow & \textsf{int ID}\{\text{record that } \textsf{ID}.\text{lexeme is of type } \textsf{int}\} \\
D & \rightarrow & \textsf{bool ID}\{\text{record that } \textsf{ID}.\text{lexeme is of type } \textsf{bool}\} \\
E & \rightarrow & E_1 + E_2\{\text{check that } E_1.\text{type} = E_2.\text{type} = \textsf{int}; \text{set } E.\text{type } := \textsf{int}\} \\
E & \rightarrow & !E_1\{\text{check that } E_1.\text{type} = \textsf{bool};\ \text{set } E.\text{type} := \textsf{bool}\} \\
E & \rightarrow & \textsf{ID}\{\text{set } E.\text{type } := \textsf{int}\}
\end{array}
$$

With respect to the above grammar, which one of the following choices is correct?

A. The actions can be used to correctly type-check any syntactically correct program
B. The actions can be used to type-check syntactically correct integer variable declarations and integer expressions
C. The actions can be used to type-check syntactically correct boolean variable declarations and boolean expressions.
D. The actions will lead to an infinite loop

gate2021-cse-set1 compiler-design syntax-directed-translation

Answer

Answers: Syntax Directed Translation

2.19.1 Syntax Directed Translation: GATE CSE 1992 | Question: 11a top

https://gateoverflow.in/590

✔

PRODUCTION RULE	SEMANTIC ACTIONS
$D \rightarrow TL$	$L.in := T.type$
$T \rightarrow int$	$T.type := integer$
$T \rightarrow real$	$T.type := real$
$L \rightarrow L, id$	$L1.in = L.in$ $Enter\_type(id.entry, L.in)$
$L \rightarrow id$	$Enter\_type(id.entry, L.in)$

15 votes

-- Anupoju Satish Kumar (325 points)

2.19.2 Syntax Directed Translation: GATE CSE 1995 | Question: 2.10 top

https://gateoverflow.in/2622

✔ Making a tree and performing post-order traversal will yield an answer as A.

- $S \rightarrow xxW$ (print "1")
- $W \rightarrow Sz$ (print "3")
- $S \rightarrow xxW$ (print "1")
- $W \rightarrow Sz$ (print "3")
- $S \rightarrow y$ (print "2")

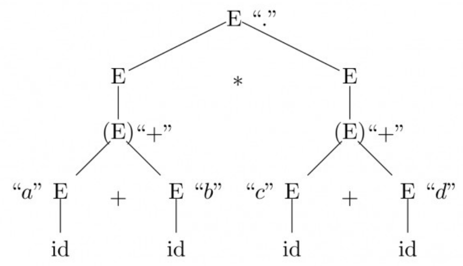
24 votes

-- Gate Keeda (15.9k points)

2.19.3 Syntax Directed Translation: GATE CSE 1996 | Question: 20 top

https://gateoverflow.in/2772

✔ $\mathbf{ab + cd+}$.



👍 25 votes -- jayendra (6.7k points)

2.19.4 Syntax Directed Translation: GATE CSE 1998 | Question: 23 top

☛ https://gateoverflow.in/1738

✔ $S \to L.L \quad \{S.val = L_1.val + L_2.val/2^{L_2.nb}\}$

$\quad | L \quad \{S.val = L.val\}$

$L \to LB \quad \{L.val = 2 * L_1.val + B.val,$
$\quad L.nb = L_1.nb + B.nb\}$

$\quad | B \quad \{L.val = B.val$
$\quad L.nb = B.nb\}$

$B \to 0 \quad \{B.val = 0$
$\quad B.nb = 1\}$

$\quad | 1 \quad \{B.val = 1$
$\quad B.nb = 1\}$

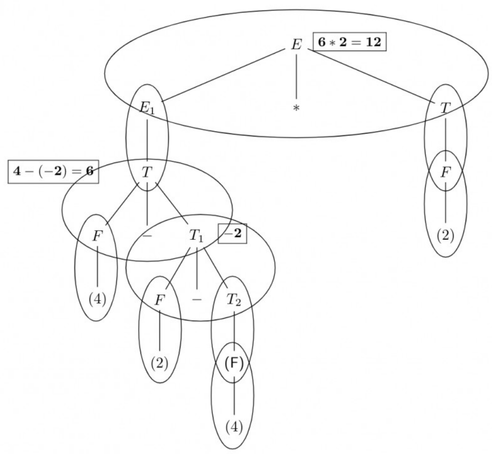
Here, val = decimal value, nb = number of bits.

👍 30 votes -- Gate Keeda (15.9k points)

2.19.5 Syntax Directed Translation: GATE CSE 2000 | Question: 19 top

☛ https://gateoverflow.in/690

✔ Given expression $4 - 2 - 4 * 2$



Total reductions = 10

a. Expression value, $E.val = 12$
b. Total number of reductions performed, $E.red = 10$ (number of non-leaf nodes in the parse tree)

👍 25 votes -- Prateek kumar (6.7k points)

2.19.6 Syntax Directed Translation: GATE CSE 2001 | Question: 17 top

☛ https://gateoverflow.in/758

✔ The desired code sequence is

```
S.begin:
    S1.code;
    E.code;
    if(E.varName = 1) goto S.next;
    goto S.begin;
S.next:
```

The following syntax-directed translation can achieve this (As mentioned in question "\\" is used to concatenate two strings)

- S.begin := newlabel()
- S.next := newlabel()
- S.code := gen(S.begin:) \\ S1.code \\ E.code \\ gen(if(E.varName = 1 goto S.next)) \\ gen(goto S.begin) \\ gen(S.next:)

5 votes -- Arjun Suresh (330k points)

2.19.7 Syntax Directed Translation: GATE CSE 2003 | Question: 18 top https://gateoverflow.in/908

✔

A. is false. If the grammar is not L-attributed; we cannot evaluate the inherited attributes in a bottom-up parse. In fact even for some L-attributed grammar, bottom-up parse is not possible for inherited attributes.
http://infolab.stanford.edu/~ullman/dragon/slides2.pdf
https://gateoverflow.in/?qa=blob&qa_blobid=14587629398289520039

B. is true. Is there any non L-attributed grammar which can be parsed by a bottom-up parser? No, as shown in the above link. In fact only for the L-attributed grammar made from a LL(1) grammar, we can always guarantee a bottom-up parsing. Even for LR(1) grammar, bottom-up parsing is not a guarantee for all inherited attributes.

C. is false. Some L-attributed grammars (including those with no synthesized attributes) can be evaluated by a bottom-up parser.

D. is false for above-told reasons.

A nice PDF for the same :- https://acm.sjtu.edu.cn/w/images/a/a1/Compiler2013-lec07.pdf

References

30 votes -- Arjun Suresh (330k points)

2.19.8 Syntax Directed Translation: GATE CSE 2016 Set 1 | Question: 46 top https://gateoverflow.in/39700

✔ **aab** could be derived as follows by the **bottom up parser:**

$S \rightarrow a\mathbf{A}$ prints 1
$\rightarrow a\mathbf{S}b$ prints 3
$\rightarrow aab$ prints 2
Now since bottom up parser will work in reverse of right most derivation, so it will print in bottom up fashion i.e., 231 which is option **C**.

Note that this can be easily visualized by drawing the derivation tree.

51 votes -- Monanshi Jain (7k points)

2.19.9 Syntax Directed Translation: GATE CSE 2019 | Question: 36 top https://gateoverflow.in/302812

✔ A node in a parse tree can INHERIT an attribute either from its parent or its siblings. This means for a production

$$S \rightarrow AB,$$

A can inherit values from either S or B and similarly B can inherit values from either S or A.

In the given productions, for

$L \rightarrow L_1, id,$

L_1 can inherit from L or , or id with only L being a non-terminal.

So, this means X_3 must be L_1 and X_4 must be L as X_i is a placeholder for non-terminals.

Only **option A** matches this.

21 votes -- Arjun Suresh (330k points)

2.19.10 Syntax Directed Translation: GATE CSE 2020 | Question: 33 top https://gateoverflow.in/333198

✔ Answer : B.

In L-attributed definitions,

1. A parent can take its attribute values from any child (which is $S-$attributed and Every $S-$attributed is also $L-$Attributed).
2. A child can take its attribute values from the parent as well as from any left sibling but not from any right sibling.

Based on these properties, only Rule-1 is $L-$attributed.
Rule-2 is failed for the production $A \rightarrow XY$, and defintion $X.i = A.i + Y.s$ since X take value from its sibling Y, which is present in its right in the production.

8 votes -- Anshul Sharma (525 points)

2.19.11 Syntax Directed Translation: GATE CSE 2021 Set 1 | Question: 26 top https://gateoverflow.in/357425

The Correct answer is **B**.

As the given grammar does type checking and also it syntactically checks integer expressions.

2 votes -- Sherrinford03 (635 points)

2.20 Target Code Generation (4) top

2.20.1 Target Code Generation: GATE CSE 1997 | Question: 4.9 top https://gateoverflow.in/2250

The expression $(a * b) * c \; op \ldots$

where 'op' is one of '$+$', '$*$' and '$\uparrow$' (exponentiation) can be evaluated on a CPU with single register without storing the value of $(a * b)$ if

A. 'op' is '$+$' or '$*$'
B. 'op' is '$\uparrow$' or '$*$'
C. 'op' is '$\uparrow$' or '$+$'
D. not possible to evaluate without storing

gate1997 compiler-design target-code-generation register-allocation normal

Answer

2.20.2 Target Code Generation: GATE CSE 2003 | Question: 59 top https://gateoverflow.in/947

Consider the syntax directed definition shown below.

$$\begin{array}{ll} S \rightarrow \textbf{id} := E & \{gen(\textbf{id}.place = E.place;);\} \\ E \rightarrow E_1 + E_2 & \{t = newtemp(); \\ & gen(t = E_1.place + E_2.place;); \\ & E.place = t;\} \\ E \rightarrow id & \{E.place = \textbf{id}.place;\} \end{array}$$

Here, gen is a function that generates the output code, and $newtemp$ is a function that returns the name of a new temporary variable on every call. Assume that t_i's are the temporary variable names generated by $newtemp$. For the statement '$X := Y + Z$', the 3-address code sequence generated by this definition is

A. $X = Y + Z$
B. $t_1 = Y + Z; X = t_1$
C. $t_1 = Y; t_2 = t_1 + Z; X = t_2$
D. $t_1 = Y; t_2 = Z; t_3 = t_1 + t_2; X = t_3$

gate2003-cse compiler-design target-code-generation normal

Answer

2.20.3 Target Code Generation: GATE CSE 2004 | Question: 10 top https://gateoverflow.in/4069

Consider the grammar rule $E \rightarrow E1 - E2$ for arithmetic expressions. The code generated is targeted to a CPU having a single user register. The subtraction operation requires the first operand to be in the register. If $E1$ and $E2$ do not have any common sub expression, in order to get the shortest possible code

A. $E1$ should be evaluated first
B. $E2$ should be evaluated first
C. Evaluation of $E1$ and $E2$ should necessarily be interleaved
D. Order of evaluation of $E1$ and $E2$ is of no consequence

gate2004-cse compiler-design target-code-generation normal

Answer

2.20.4 Target Code Generation: GATE CSE 2010 | Question: 37 top https://gateoverflow.in/2338

The program below uses six temporary variables a, b, c, d, e, f.

```
a = 1
b = 10
c = 20
d = a + b
e = c + d
f = c + e
b = c + e
e = b + f
d = 5 + e
return d + f
```

Assuming that all operations take their operands from registers, what is the minimum number of registers needed to execute this program without spilling?

A. 2
B. 3
C. 4
D. 6

gate2010-cse compiler-design target-code-generation register-allocation normal

Answer

Answers: Target Code Generation

2.20.1 Target Code Generation: GATE CSE 1997 | Question: 4.9 top https://gateoverflow.in/2250

✔ Correct Option: A

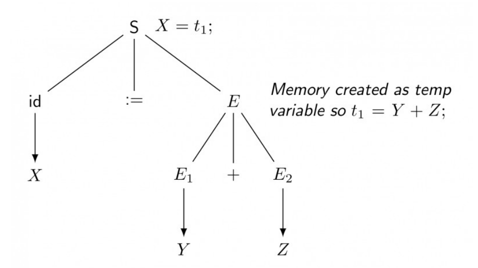
$\uparrow$ has higer precedence than $\{*, +, -, /\}$

So, if op $= \uparrow$ implies, we need to evaluate the right hand side of $\uparrow$ first and then do the lhs part, which would definitely require us to store the value of lhs

but if its a '$+$' or '$*$' , we don't need to store the values evaluated, and on the go can do the operation directly on one register.

25 votes -- confused_luck (741 points)

2.20.2 Target Code Generation: GATE CSE 2003 | Question: 59 top https://gateoverflow.in/947



Answer (B)

29 votes -- Prashant Singh (47.1k points)

2.20.3 Target Code Generation: GATE CSE 2004 | Question: 10 top https://gateoverflow.in/4069

$E2$ should be evaluated first

After evaluating $E2$ first and then $E1$, we will have $E1$ in the register and thus we can simply do SUB operation with $E2$ which will be in memory (as we have only a single register). If we do $E1$ first and then $E2$, we must move $E2$ to memory and $E1$ back to register before doing SUB, which will increase the code size.

60 votes -- Arjun Suresh (330k points)

2.20.4 Target Code Generation: GATE CSE 2010 | Question: 37 top https://gateoverflow.in/2338

Here in these types of compiler questions, idea is "map/assign multiple temporaries to one registers."

here a, b, and c all are having 3 different values so i need atleast 3 registers $r1$, $r2$ and $r3$.
a is mapped to $r1$, b to $r2$ and c to $r3$.

$d = a + b$, after this line if u notice 'a' is never present on right hand side, so I can map (register of a which is $r1$) d to $r1$.
$e = c + d$, after this line 'd' is never present on rhs, so I can map (register of d which is $r1$) e to $r1$.

at this time mapping is
$r1 - - - e$
$r2 - - - b$
$r3 - - - c$

(at this moment I have registers for e, b and c. if I introduce new variable then I may need different register)
now at this point if u see
$f = c + e$
$b = c + e$
these two are essentially doing same thing, after these two line 'b' and 'f' are same so I can skip computing 'f'. and whereever f is present I will replace it with 'b'. (because neither of 'f' and 'b' are changing after these two lines, so value of these will be '$c + e$' forever)

(seems like I introduced one more variable f, and register is needed for that, but actually I did not really introduce 'f'. I am skipping computation of 'f')
now at second last line "$d = 5 + e$"
here I introduced 'd', I can map it to any of the register $r1$ or $r3$, because after this line neither of 'e' or 'c' is required. (value of 'b' is required because I need to return '$d + f$', and 'f' is essentially equal to 'b')

finally code becomes

$r1 = 1$
$r2 = 10$
$r3 = 20$
$r1 = r1 + r2$
$r1 = r3 + r1$

(skipping 'f' computation)
$r2 = r3 + r1$
$r2 = r3 + r1$
$r1 = r2 + r2$

$r3 = 5 + r1$
return $r3 + r2$

Therefore minimum 3 registers needed.

Correct Answer: B

57 votes -- Sachin Mittal (15.8k points)

2.21 Variable Scope (2) top

2.21.1 Variable Scope: GATE CSE 1987 | Question: 1-xix top https://gateoverflow.in/80373

Study the following program written in a block-structured language:

```
Var x, y:interger;
procedure P(n:interger);
begin
     x:=(n+2)/(n-3);
end;

procedure Q
Var x, y:interger;
begin
    x:=3;
    y:=4;
    P(y);
    Write(x)                                    __(1)
end;

begin
    x:=7;
    y:=8;
    Q;
Write(x);                                       __(2)
end.
```

What will be printed by the write statements marked (1) and (2) in the program if the variables are statically scoped?

A. $3, 6$
B. $6, 7$
C. $3, 7$
D. None of the above.

gate1987 compiler-design variable-scope runtime-environments

Answer

2.21.2 Variable Scope: GATE CSE 1987 | Question: 1-xx top https://gateoverflow.in/80374

For the program given below what will be printed by the write statements marked (1) and (2) in the program if the variables are dynamically scoped?

```
Var x, y:interger;
procedure P(n:interger);
begin
     x := (n+2)/(n-3);
end;

procedure Q
Var x, y:interger;
begin
    x:=3;
    y:=4;
    P(y);
    Write(x);                                    __(1)
end;

begin
    x:=7;
    y:=8;
    Q;
    Write(x);                                       __(2)
end.
```

A. $3, 6$
B. $6, 7$

C. 3, 7
D. None of the above

gate1987 compiler-design variable-scope runtime-environments

Answer

Answers: Variable Scope

2.21.1 Variable Scope: GATE CSE 1987 | Question: 1-xix top https://gateoverflow.in/80373

✔ **Using Static Scoping:**
First, procedure **Q** is called from the **main** procedure. **Q** has local variables **x** and **y** with values **3** and **4** respectively. This local variable **y** (value **4**) is being passed to procedure **P** during call, and received in local variable **n** inside procedure **P**. Now, as **P** does not have any local definition for variable **x**, it will assign the evaluated value of **(n+2)/(n-3)** i.e. **(4+2)/(4-3)=6** to the global variable **x**, which was previously **7**. After the call of procedure **P**, procedure **Q** writes the value of local variable **x** which is still **3**. Lastly, the **main** procedure writes the value of global variable **x** which has been changed to **6** inside procedure **P**. So, the output will be **3, 6**.

Using Dynamic Scoping:
The same sequence of statements will be executed using dynamic scoping. However, as there is no local definition of variable **x** in procedure **P**, it will consider the recent definition in the calling sequence; as **P** is being called from procedure **Q**, definition of **x** from **Q** will be used, and value of **x** will be changed to **6** from **3**. Now, when **Q** writes local variable **x**, **6** will be printed. The write global variable **x** from **main** procedure will print **7** (as value of the global variable **x** has not been changed). So, the output will be **6, 7**.

Correct Answer: A

35 votes -- Sutanay Bhattacharjee (3.1k points)

2.21.2 Variable Scope: GATE CSE 1987 | Question: 1-xx top https://gateoverflow.in/80374

✔ **Using Static Scoping:**
First, procedure **Q** is called from the **main** procedure. **Q** has local variables **x** and **y** with values **3** and **4** respectively. This local variable **y** (value **4**) is being passed to procedure **P** during call, and received in local variable **n** inside procedure **P**. Now, as **P** does not have any local definition for variable **x**, it will assign the evaluated value of **(n+2)/(n-3)** i.e. **(4+2)/(4-3)=6** to the global variable **x**, which was previously **7**. After the call of procedure **P**, procedure **Q** writes the value of local variable **x** which is still **3**. Lastly, the **main** procedure writes the value of global variable **x** which has been changed to **6** inside procedure **P**. So, the output will be **3, 6**.

Using Dynamic Scoping:
The same sequence of statements will be executed using dynamic scoping. However, as there is no local definition of variable **x** in procedure **P**, it will consider the recent definition in the calling sequence; as **P** is being called from procedure **Q**, definition of **x** from **Q** will be used, and value of **x** will be changed to **6** from **3**. Now, when **Q** writes local variable **x**, **6** will be printed. The write global variable **x** from **main** procedure will print **7** (as value of the global variable **x** has not been changed). So, the output will be **6, 7**.

Correct Answer: B

10 votes -- Sutanay Bhattacharjee (3.1k points)

2.22 Viable Prefix (1) top

2.22.1 Viable Prefix: GATE CSE 2015 Set 1 | Question: 13 top https://gateoverflow.in/8187

Which one of the following is TRUE at any valid state in shift-reduce parsing?

A. Viable prefixes appear only at the bottom of the stack and not inside
B. Viable prefixes appear only at the top of the stack and not inside
C. The stack contains only a set of viable prefixes
D. The stack never contains viable prefixes

gate2015-cse-set1 compiler-design parsing normal viable-prefix

Answer

Answers: Viable Prefix

2.22.1 Viable Prefix: GATE CSE 2015 Set 1 | Question: 13 top

https://gateoverflow.in/8187

✔ **Answer - C**

Explanation:

A handle is actually the one which is always on the top of the stack. A viable prefix(prefix of the Right-hand side of a production or productions), is actually a prefix of the handle and so can never extend past the right end of the handle(i.e. the top of the stack).

The structure of the stack can be considered as a set of viable prefixes -

$Stack = \{Prefix_1 Prefix_2 Prefix_3 \dots Prefix_{n-1} Prefix_n\}$ and so it is not wrong to say that the stack contains a set of viable prefixes.

61 votes -- Ravi Ranjan (3k points)

Answer Keys

2.1.1	C	2.2.1	N/A	2.2.2	N/A	2.2.3	N/A	2.2.4	A;B;C
2.2.5	True	2.2.6	N/A	2.2.7	True	2.2.8	False	2.2.9	B
2.3.1	A	2.3.2	D	2.3.3	7	2.3.4	A	2.3.5	6 : 6
2.3.6	D	2.3.7	A	2.4.1	C	2.4.2	N/A	2.4.3	A
2.4.4	B	2.4.5	C	2.4.6	B	2.4.7	C	2.4.8	B
2.4.9	D	2.4.10	C	2.5.1	A	2.5.2	6	2.6.1	N/A
2.6.2	N/A	2.6.3	N/A	2.6.4	N/A	2.6.5	C	2.6.6	N/A
2.6.7	N/A	2.6.8	D	2.6.9	N/A	2.6.10	N/A	2.6.11	D
2.6.12	A	2.6.13	N/A	2.6.14	N/A	2.6.15	N/A	2.6.16	C
2.6.17	B	2.6.18	A	2.6.19	N/A	2.6.20	D	2.6.21	B
2.6.22	C	2.6.23	B	2.6.24	C	2.6.25	D	2.6.26	B
2.6.27	D	2.6.28	D	2.6.29	A	2.6.30	C	2.6.31	C
2.6.32	C	2.6.33	B	2.6.34	C	2.6.35	C	2.6.36	B
2.6.37	B	2.6.38	A	2.6.39	C	2.6.40	5	2.6.41	A
2.6.42	C	2.7.1	N/A	2.7.2	N/A	2.7.3	N/A	2.7.4	C
2.7.5	A	2.7.6	C	2.7.7	8	2.7.8	A	2.7.9	B
2.7.10	D	2.8.1	A	2.8.2	C	2.8.3	B	2.8.4	C
2.8.5	A	2.8.6	D	2.9.1	B	2.9.2	C	2.9.3	C
2.10.1	C	2.11.1	C	2.12.1	7	2.12.2	8 : 8	2.13.1	N/A
2.13.2	D	2.13.3	A	2.13.4	D	2.14.1	10	2.14.2	N/A
2.14.3	A;D	2.14.4	N/A	2.14.5	A;D	2.14.6	N/A	2.14.7	N/A
2.14.8	N/A	2.14.9	A	2.14.10	N/A	2.14.11	X	2.14.12	D
2.14.13	D	2.14.14	D	2.15.1	A	2.15.2	N/A	2.15.3	N/A
2.15.4	N/A	2.15.5	A;B	2.15.6	B;C;D	2.15.7	N/A	2.15.8	N/A
2.15.9	C	2.15.10	C	2.15.11	N/A	2.15.12	B	2.15.13	A
2.15.14	N/A	2.15.15	N/A	2.15.16	D	2.15.17	B	2.15.18	A
2.15.19	A	2.15.20	B	2.15.21	C	2.15.22	B	2.15.23	A
2.15.24	D	2.15.25	A	2.15.26	D	2.15.27	B	2.15.28	B

2.15.29	B	2.15.30	A	2.15.31	C	2.15.32	D	2.15.33	B
2.15.34	D	2.15.35	C	2.15.36	D	2.15.37	9	2.15.38	C
2.15.39	1024	2.15.40	A	2.15.41	A	2.15.42	31	2.15.43	D
2.15.44	A	2.15.45	A	2.15.46	A	2.16.1	D	2.16.2	2
2.17.1	N/A	2.17.2	N/A	2.17.3	N/A	2.17.4	N/A	2.17.5	True
2.17.6	D	2.17.7	C	2.17.8	B	2.17.9	C	2.17.10	C
2.17.11	B	2.17.12	C	2.17.13	C	2.17.14	A	2.17.15	C
2.17.16	D	2.17.17	A	2.17.18	D	2.17.19	C	2.18.1	10
2.18.2	B	2.19.1	N/A	2.19.2	A	2.19.3	N/A	2.19.4	N/A
2.19.5	N/A	2.19.6	N/A	2.19.7	B	2.19.8	C	2.19.9	A
2.19.10	B	2.19.11	B	2.20.1	A	2.20.2	B	2.20.3	B
2.20.4	B	2.21.1	A	2.21.2	B	2.22.1	C		

3 Programming and DS: DS (213)

Arrays, Stacks, Queues, Linked lists, Trees, Binary search trees, Binary heaps, Graphs.

Mark Distribution in Previous GATE

Year	2021-1	2021-2	2020	2019	2018	2017-1	2017-2	2016-1	2016-2	Minimum	Average	Maximum
1 Mark Count	4	2	2	0	2	3	1	1	1	0	1.7	4
2 Marks Count	1	0	1	2	0	0	1	3	3	0	1.2	3
Total Marks	6	2	4	4	2	3	3	7	7	**2**	**4.2**	**7**

3.1 Abstract Data Type (1) top

3.1.1 Abstract Data Type: GATE CSE 2005 | Question: 2 top https://gateoverflow.in/1344

An Abstract Data Type (ADT) is:

A. same as an abstract class
B. a data type that cannot be instantiated
C. a data type for which only the operations defined on it can be used, but none else
D. all of the above

gate2005-cse data-structures normal abstract-data-type

Answer

Answers: Abstract Data Type

3.1.1 Abstract Data Type: GATE CSE 2005 | Question: 2 top https://gateoverflow.in/1344

✔ An abstract data type (ADT) supports only the operations which are defined.

Abstract class is one that may not have definitions of all the objects it have. Moreover it can not be instantiated. To instantiate we have to create a subclass then instantiate the class.

Abstract Data Type is like data structure eg. $STACK$ where we have $PUSH()$ $POP()$ operation defined .

Hence, they are not the same thing.

http://www.devx.com/tips/Tip/5681

Correct Answer: C

References

49 votes -- Manali (2.1k points)

3.2 Arrays (13) top

3.2.1 Arrays: GATE CSE 1993 | Question: 12 top https://gateoverflow.in/2309

The following Pascal program segments finds the largest number in a two-dimensional integer array $A[0\dots n-1, 0\dots n-1]$ using a single loop. Fill up the boxes to complete the program and write against $\boxed{\text{A}}, \boxed{\text{B}}, \boxed{\text{C}}$ and $\boxed{\text{D}}$ in your answer book Assume that max is a variable to store the largest value and i, j are the indices to the array.

```
begin
    max:=|A|, i:=0, j:=0;
    while |B| do
    begin
        if A[i, j]>max then max:=A[i, j];
        if |C| then j:=j+1;
        else begin
            j:=0;
            i:=|D|
        end
    end
```

```
end
```

gate1993 data-structures arrays normal descriptive

Answer

3.2.2 Arrays: GATE CSE 1994 | Question: 1.11 top https://gateoverflow.in/2452

In a compact single dimensional array representation for lower triangular matrices (i.e all the elements above the diagonal are zero) of size $n \times n$, non-zero elements, (i.e elements of lower triangle) of each row are stored one after another, starting from the first row, the index of the $(i, j)^{th}$ element of the lower triangular matrix in this new representation is:

A. $i + j$
B. $i + j - 1$
C. $(j - 1) + \frac{i(i-1)}{2}$
D. $i + \frac{j(j-1)}{2}$

gate1994 data-structures arrays normal

Answer

3.2.3 Arrays: GATE CSE 1994 | Question: 25 top https://gateoverflow.in/2521

An array A contains n integers in non-decreasing order, $A[1] \leq A[2] \leq \cdots \leq A[n]$. Describe, using Pascal like pseudo code, a linear time algorithm to find i, j, such that $A[i] + A[j] =$ a given integer M, if such i, j exist.

gate1994 data-structures arrays normal descriptive

Answer

3.2.4 Arrays: GATE CSE 1997 | Question: 17 top https://gateoverflow.in/2277

An array A contains $n \geq 1$ positive integers in the locations $A[1], A[2], \ldots A[n]$. The following program fragment prints the length of a shortest sequence of consecutive elements of A, $A[i], A[i + 1], \ldots, A[j]$ such that the sum of their values is $\geq M$, a given positive number. It prints ‘ $n + 1$’ if no such sequence exists. Complete the program by filling in the boxes. In each case use the simplest possible expression. Write only the line number and the contents of the box.

```
begin
i:=1;j:=1;
sum := □
min:=n; finish:=false;
while not finish do
    if □ then
        if j=n then finish:=true
        else
        begin
            j:=j+1;
            sum:= □
        end
    else
    begin
        if(j-i) < min then min:=j-i;
        sum:=sum -A[i];
        i:=i+1;
    end
    writeln (min +1);
end.
```

gate1997 data-structures arrays normal descriptive

Answer

3.2.5 Arrays: GATE CSE 1998 | Question: 2.14 top https://gateoverflow.in/1686

Let A be a two dimensional array declared as follows:

```
A: array [1 …. 10] [1 ….. 15] of integer;
```

Assuming that each integer takes one memory location, the array is stored in row-major order and the first element of the array is stored at location 100, what is the address of the element $A[i][j]$?

A. $15i + j + 84$

B. $15j + i + 84$
C. $10i + j + 89$
D. $10j + i + 89$

gatel998 data-structures arrays easy

Answer

3.2.6 Arrays: GATE CSE 2000 | Question: 1.2 top

https://gateoverflow.in/625

An $n \times n$ array v is defined as follows:

$v[i, j] = i - j$ for all $i, j, i \leq n, 1 \leq j \leq n$

The sum of the elements of the array v is

A. 0
B. $n - 1$
C. $n^2 - 3n + 2$
D. $n^2 \frac{(n+1)}{2}$

gate2000-cse data-structures arrays easy

Answer

3.2.7 Arrays: GATE CSE 2000 | Question: 15 top

https://gateoverflow.in/686

Suppose you are given arrays $p[1......N]$ and $q[1......N]$ both uninitialized, that is, each location may contain an arbitrary value), and a variable count, initialized to 0. Consider the following procedures set and is_set:

```
set(i) {
    count = count + 1;
    q[count] = i;
    p[i] = count;
}
is_set(i) {
    if (p[i] ≤ 0 or p[i] > count)
        return false;
    if (q[p[i]] ≠ i)
        return false;
    return true;
}
```

A. Suppose we make the following sequence of calls:
   $set(7); set(3); set(9);$
   After these sequence of calls, what is the value of count, and what do $q[1], q[2], q[3], p[7], p[3]$ and $p[9]$ contain?
B. Complete the following statement "The first count elements of __________contain values i such that set (______________) has been called".
C. Show that if $set(i)$ has not been called for some i, then regardless of what $p[i]$ contains, $is\_set(i)$ will return false.

gate2000-cse data-structures arrays easy descriptive

Answer

3.2.8 Arrays: GATE CSE 2005 | Question: 5 top

https://gateoverflow.in/1347

A program P reads in 500 integers in the range $[0, 100]$ representing the scores of 500 students. It then prints the frequency of each score above 50. What would be the best way for P to store the frequencies?

A. An array of 50 numbers
B. An array of 100 numbers
C. An array of 500 numbers
D. A dynamically allocated array of 550 numbers

gate2005-cse data-structures arrays easy

Answer

3.2.9 Arrays: GATE CSE 2013 | Question: 50 top

https://gateoverflow.in/1557

The procedure given below is required to find and replace certain characters inside an input character string supplied in array A. The characters to be replaced are supplied in array $oldc$, while their respective replacement characters are supplied in array $newc$. Array A has a fixed length of five characters, while arrays $oldc$ and $newc$ contain three characters each. However, the procedure is flawed.

```
void find_and_replace (char *A, char *oldc, char *newc) {
    for (int i=0; i<5; i++)
        for (int j=0; j<3; j++)
            if (A[i] == oldc[j])
                A[i] = newc[j];
}
```

The procedure is tested with the following four test cases.

1. $oldc =$ "abc", $newc =$ "dab"
2. $oldc =$ "cde", $newc =$ "bcd"
3. $oldc =$ "bca", $newc =$ "cda"
4. $oldc =$ "abc", $newc =$ "bac"

The tester now tests the program on all input strings of length five consisting of characters 'a', 'b', 'c', 'd' and 'e' with duplicates allowed. If the tester carries out this testing with the four test cases given above, how many test cases will be able to capture the flaw?

A. Only one
B. Only two
C. Only three
D. All four

gate2013-cse data-structures arrays normal

Answer

3.2.10 Arrays: GATE CSE 2013 | Question: 51 top

https://gateoverflow.in/43291

The procedure given below is required to find and replace certain characters inside an input character string supplied in array A. The characters to be replaced are supplied in array $oldc$, while their respective replacement characters are supplied in array $newc$. Array A has a fixed length of five characters, while arrays $oldc$ and $newc$ contain three characters each. However, the procedure is flawed.

```
void find_and_replace (char *A, char *oldc, char *newc) {
    for (int i=0; i<5; i++)
        for (int j=0; j<3; j++)
            if (A[i] == oldc[j])
                A[i] = newc[j];
}
```

The procedure is tested with the following four test cases.

1. $oldc =$ "abc", $newc =$ "dab"
2. $oldc =$ "cde", $newc =$ "bcd"
3. $oldc =$ "bca", $newc =$ "cda"
4. $oldc =$ "abc", $newc =$ "bac"

If array A is made to hold the string "$abcde$", which of the above four test cases will be successful in exposing the flaw in this procedure?

A. None
B. 2 only
C. 3 and 4 only
D. 4 only

gate2013-cse data-structures arrays normal

Answer

3.2.11 Arrays: GATE CSE 2014 Set 3 | Question: 42 top https://gateoverflow.in/2076

Consider the C function given below. Assume that the array $listA$ contains $n(>0)$ elements, sorted in ascending order.

```
int ProcessArray(int *listA, int x, int n)
{
        int i, j, k;
        i = 0;      j = n-1;
         do {
               k = (i+j)/2;
               if (x <= listA[k]) j = k-1;
               if (listA[k] <= x) i = k+1;
            }
        while (i <= j);
        if (listA[k] == x) return(k);
        else   return -1;
}
```

Which one of the following statements about the function $ProcessArray$ is **CORRECT**?

A. It will run into an infinite loop when x is not in $listA$.
B. It is an implementation of binary search.
C. It will always find the maximum element in $listA$.
D. It will return -1 even when x is present in $listA$.

gate2014-cse-set3 data-structures arrays easy

Answer

3.2.12 Arrays: GATE CSE 2015 Set 2 | Question: 31 top https://gateoverflow.in/8148

A Young tableau is a $2D$ array of integers increasing from left to right and from top to bottom. Any unfilled entries are marked with ∞, and hence there cannot be any entry to the right of, or below a ∞. The following Young tableau consists of unique entries.

1	2	5	14
3	4	6	23
10	12	18	25
31	∞	∞	∞

Legendary q

When an element is removed from a Young tableau, other elements should be moved into its place so that the resulting table is still a Young tableau (unfilled entries may be filled with a ∞). The minimum number of entries (other than 1) to be shifted, to remove 1 from the given Young tableau is _____.

gate2015-cse-set2 databases arrays normal numerical-answers

Answer

3.2.13 Arrays: GATE CSE 2021 Set 1 | Question: 2 top https://gateoverflow.in/357450

Let P be an array containing n integers. Let t be the lowest upper bound on the number of comparisons of the array elements, required to find the minimum and maximum values in an arbitrary array of n elements. Which one of the following choices is correct?

A. $t > 2n-2$
B. $t > 3\lceil\frac{n}{2}\rceil$ and $t \leq 2n-2$
C. $t > n$ and $t \leq 3\lceil\frac{n}{2}\rceil$
D. $t > \lceil\log_2(n)\rceil$ and $t \leq n$

gate2021-cse-set1 data-structures arrays

Answer

Answers: Arrays

3.2.1 Arrays: GATE CSE 1993 | Question: 12 top https://gateoverflow.in/2309

✔ We have to traverse all elements in array. The code is doing this row wise.

```
begin
    max:=A[0,0], i:=0, j:=0;
    while (i < n) do
    begin
        if A[i, j]>max then max:=A[i, j];
        if (j < n-1) then j:=j+1;
        else begin
            j:=0;
            i:=i++;
        end
    end
end
```

36 votes -- Arjun Suresh (330k points)

3.2.2 Arrays: GATE CSE 1994 | Question: 1.11 top https://gateoverflow.in/2452

✔ $j-1+i(i-1)/2$ because if you form a lower triangular matrix it contains elements in rows $1,2,3,\dots$

So, **C** is the correct answer.

PS: Though not mentioned in the question, from options it is clear that the **array index starts from 1 and not** 0.

Explanation :

In a lower triangular matrix, i^{th} row contains $(i+1)$ number of non zero elements.

If we assume Array index starting from 1 then, i^{th} row contains i number of non zero elements.

before i^{th} row there are $i-1$ rows (row 1 to $i-1$) , and in total these rows has $1+2+3+\ldots+(i-1)=i((i-1)/2$ elements (row 1 has 1 element, row 2 has 2 elements, row $i-1$ has $i-1$ elements etc.)

Now, at i^{th} row, before j^{th} element there are $(j-1)$ elements(starting from $arr[i,1]$ to $arr[i,j-1]$) .

Hence, in total before $arr[i,j]$ there are $(i(i-1)/2+j-1)$ elements and those elements will have indexes .

S,o the index of the $(i,j)^{th}$ element of the lower triangular matrix in this new representation is $(j-1)+i(i-1)/2$ which is option **C** .

70 votes -- Bhagirathi Nayak (11.7k points)

3.2.3 Arrays: GATE CSE 1994 | Question: 25 top https://gateoverflow.in/2521

✔

```
   i = 1;
j = n;
while(i != j) {
   if(A[i] + A[j] == M) break;
   else if(A[i] + A[j] < M) i++;
   else j--;
}
```

35 votes -- Ankit Rokde (6.9k points)

3.2.4 Arrays: GATE CSE 1997 | Question: 17 top https://gateoverflow.in/2277

✔
```
begin
i:=1;j:=1;
sum := A[1]
min:=n; finish:=false;
while not finish do
    if sum < M then
        if j=n then finish:=true
        else
        begin
            j:=j+1;
            sum:= sum + A[j]
        end
    else
    begin
        if(j-i) < min then min:=j-i;
        sum:=sum – A[i];
        i:=i+1;
    end
    writeln (min +1);
end.
```

Algorithm

'i' indicates the starting marker and 'j' acts as ending marker for the sum sequence. 'sum' is initialised as the first element in the array because the algorithm proceeds by taking the sum of remaining elements. 'finish' is a boolean variable that indicates exit from the loop.

After entering the loop for the first time with ' $finish$' as false, the sum is checked if it's strictly less than " M". If that's the case j is incremented and the sum is modified to sum $+A[j]$. When 'sum' becomes greater than or equal to ' M', 'min' is modified to the latest number of elements that make the sum greater than or equal to 'M' and then, the first element is stripped off from the sum and 'i' is incremented by one to move the initial marker of the sum sequence. The loop runs till ' j' reaches the end of the array.

The algorithm keeps track of 'min' i.e. the number of elements in the minimum sum sequence. This is very similar to the way we find the minimum value in an array by modifying the min value whenever a lesser value is encountered.

22 votes -- krish__ (4.6k points)

3.2.5 Arrays: GATE CSE 1998 | Question: 2.14 top

https://gateoverflow.in/1686

✔ $A[LB_1 \dots UB_1, LB_2 \dots UB_2]$

$BA =$ Base address.

$C =$ size of each element.

Row major order.

$Loc(a[i][j]) = BA + [(i - LB_1)(UB_2 - LB_2 + 1) + (j - LB_2)] * C.$

Column Major order

$Loc(a[i][j]) = BA + [(j - LB_2)(UB_1 - LB_1 + 1) + (i - LB_1)] * C.$

Substituting the values.

Answer is **A**.

31 votes -- Gate Keeda (15.9k points)

3.2.6 Arrays: GATE CSE 2000 | Question: 1.2 top

https://gateoverflow.in/625

✔ The sum of the i^{th} row and i^{th} column is 0 as shown below. Since, the numbers of rows equals the number of columns, the total sum will be 0.

0	-1	-2	-3	-4
1	0	-1	-2	-3
2	1	0	-1	-2
3	2	1	0	-1
4	3	2	1	0

Correct Answer: A

46 votes -- Arjun Suresh (330k points)

3.2.7 Arrays: GATE CSE 2000 | Question: 15 top https://gateoverflow.in/686

✔

a.

Initially $count = 0$;

When we call $set(7) - count = 1, q[1] = 7, p[7] = 1$;
when we call $set(3) - count = 2, q[2] = 3, p[3] = 2$;
when we call $set(9) - count = 3, q[3] = 9, p[9] = 3$;

b. Ans- "The first count elements of $array\ q$ contain values i such that $set(i)$ has been called".

c. If $set(i)$ has not been called for some i, then regardless of what $p[i]$ contains, When we call $is\_set(i)$ then

```
if (q[p[i]] ≠ i)
      return false;
```

will always execute, because if set(i) is not called then p[i] ≠ count(any) and for then same count q[count] ≠ i. So if statement will be true and will return false.

16 votes -- Dhananjay Kumar Sharma (18.8k points)

3.2.8 Arrays: GATE CSE 2005 | Question: 5 top https://gateoverflow.in/1347

✔ As we our area of interest is only the 50 numbers so take An array of 50 numbers where $A[0]$ corresponds to $51 \ldots A[49]$ corresponds to 100 then after reading an input just increment the counter in correct position as said above.

Correct Answer: A

63 votes -- Bhagirathi Nayak (11.7k points)

3.2.9 Arrays: GATE CSE 2013 | Question: 50 top https://gateoverflow.in/1557

✔ The test cases 3 and 4 are the only cases that capture the flaw. The code does not work properly when an old character is replaced by a new character and the new character is again replaced by another new character. This doesn't happen in test cases (1) and (2), it happens only in cases (3) and (4).

Correct Answer: B.

42 votes -- Vikrant Singh (11.2k points)

3.2.10 Arrays: GATE CSE 2013 | Question: 51 top https://gateoverflow.in/43291

✔ The test cases 3 and 4 are the only cases that capture the flaw. The code does not work properly when an old character is replaced by a new character and the new character is again replaced by another new character. This does not happen in test cases (1) and (2), it happens only in cases (3) and (4).

Correct Answer: C.

8 votes -- Vikrant Singh (11.2k points)

3.2.11 Arrays: GATE CSE 2014 Set 3 | Question: 42 top https://gateoverflow.in/2076

✔ This is an implementation of the Binary search algorithm.

Note that the loop will be terminated when we have found x. In that case both the if conditions will be true making condition inside the while as false i.e., $i > j$.

Correct Answer: B

28 votes -- Monanshi Jain (7k points)

3.2.12 Arrays: GATE CSE 2015 Set 2 | Question: 31 top

☛ https://gateoverflow.in/8148

✔ The answer should be 5.

1. We first need to shift 2 in place of 1 keeping 5 AND 14 intact as it isn't mentioned in the question that the entire row elements move.
2. 4 is shifted up,next to 2 (keeping 12 and infinity intact in column 2).
3. Now in second row 6 is shifted left.
4. 18 shifts up to the second row
5. And finally 25 is shifted left to the third column.

So, this takes 5 moves and still maintains the tableau property. Also infinity is placed to the right of 25 and below 23 (unfilled entries to be filled with ∞). The final table would look as follows.

2	4	5	14
3	6	18	23
10	12	25	∞
31	∞	∞	∞

👍 71 votes

-- Aman verma (209 points)

3.2.13 Arrays: GATE CSE 2021 Set 1 | Question: 2 top

☛ https://gateoverflow.in/357450

✔ **Correct Option: C**

The answer is a stub!

Tournament Method:

- Imagine using **merge sort** and you'd divided the array elements in **pairs of two**, every element is compared with each other.
- The **largest**(or smallest if preferred) is selected out each **pairs** and the **winners** are copied to a new array and the procedure it repeated till we have one element remaining.

For this,
At first we need $\frac{n}{2}$ comparisons(since $\frac{n}{2}$ pairs), then $\frac{n}{4}$, so on this sums to n , an AP problem.

$$\text{Comparisons Req. for finding the Max Element} = \frac{n}{2} + \frac{n}{4} + \frac{n}{8} \ldots = n$$

- For finding the smallest element we would use the losers left out in the first round $\frac{n}{2}$ losers to be precise.
- We again use this procedure with an intention for finding smaller amongst all, (worst losers will be the best winners in these rounds, ironical indeed).

For this,
We need, $\frac{n}{4}$ at first since we are pitting losers against losers comparisons then, $\frac{n}{8}$ so on which sums upto $\frac{n}{2}$.

$$\text{Comparisons Req. for finding the Min Element} = \frac{n}{4} + \frac{n}{8} + \frac{n}{16} \ldots = \frac{n}{2}$$

$$\text{Total Number of Comparisons Number Required} = n + \frac{n}{2} = \frac{3n}{2}$$

The number of comparisons needed is at least $\frac{3}{2}n$ if we consider the elements to be distinct.

Hence **C** is the answer.

Another more intuitive way to understand this is the build heap operation. I'll leave that to you...

👍 2 votes

-- Cringe is my middle name... (817 points)

3.3 Avl Tree (3) top

3.3.1 Avl Tree: GATE CSE 2009 | Question: 37,ISRO-DEC2017-55 top

☛ https://gateoverflow.in/1323

What is the maximum height of any AVL-tree with 7 nodes? Assume that the height of a tree with a single node is 0.

A. 2
B. 3
C. 4
D. 5

gate2009-cse data-structures binary-search-tree normal isrodec2017 avl-tree

Answer

3.3.2 Avl Tree: GATE CSE 2020 | Question: 6 top https://gateoverflow.in/333225

What is the worst case time complexity of inserting n^2 elements into an AVL-tree with n elements initially?

A. $\Theta(n^4)$
B. $\Theta(n^2)$
C. $\Theta(n^2 \log n)$
D. $\Theta(n^3)$

gate2020-cse binary-tree avl-tree

Answer

3.3.3 Avl Tree: GATE IT 2008 | Question: 12 top https://gateoverflow.in/3272

Which of the following is TRUE?

A. The cost of searching an AVL tree is $\Theta(\log n)$ but that of a binary search tree is $O(n)$
B. The cost of searching an AVL tree is $\Theta(\log n)$ but that of a complete binary tree is $\Theta(n \log n)$
C. The cost of searching a binary search tree is $O(\log n)$ but that of an AVL tree is $\Theta(n)$
D. The cost of searching an AVL tree is $\Theta(n \log n)$ but that of a binary search tree is $O(n)$

gate2008-it data-structures binary-search-tree easy avl-tree

Answer

Answers: Avl Tree

3.3.1 Avl Tree: GATE CSE 2009 | Question: 37,ISRO-DEC2017-55 top https://gateoverflow.in/1323

✔ Answer is **B**.
With 1 node height is 0.

Max height will come when each level contain min nodes.

Minimum Nodes in an AVL tree with height n is $H(n) = H(n-1) + H(n-2) + 1$.
$H(0) = 1$.
$H(1) = 2$
$H(2) = H(1) + H(0) + 1 = 2 + 1 + 1 = 4$
$H(3) = H(2) + H(1) + 1 = 4 + 2 + 1 = 7$.

So, the max height with 7 nodes is 3.

71 votes -- Gate Keeda (15.9k points)

3.3.2 Avl Tree: GATE CSE 2020 | Question: 6 top https://gateoverflow.in/333225

✔ Answer : C
Steps: For **worst case** (in worst case insertion will cause $\Omega(\log n)$ operations in an AVL tree where n is the number of nodes present.

- 1^{st} insertion time complexity $= \Theta(\log n)$
- 2^{nd} insertion time complexity $= \Theta(\log(n+1))$
- $\vdots$
- n^{2th} insertion time complexity $= \Theta(\log n) + \Theta(\log(n+1)) + \ldots + \Theta(\log(n+n^2))$
$= \Theta\left(\log(n.(n+1).(n+2)\ldots(n+n^2))\right)$

$$= \Theta\left(\log n^{n^2}\right) \; (\because \text{the highest degree term in the } \log \text{ expression will be } n^{n^2}.$$
$$= \Theta\left(n^2 \log n\right).$$

👍 14 votes -- Prashant Singh (47.1k points)

3.3.3 Avl Tree: GATE IT 2008 | Question: 12 top ☛ https://gateoverflow.in/3272

✔ **A**) is true as AVL tree is a balanced search tree that has time complexity of searching $\Theta(\log n)$, but in binary search tree, we can have a completely left/right skewed tree, in which search is $O(n)$.

👍 37 votes -- Happy Mittal (8.2k points)

3.4 Binary Heap (3) top

3.4.1 Binary Heap: GATE CSE 2018 | Question: 46 top ☛ https://gateoverflow.in/204121

The number of possible min-heaps containing each value from $\{1, 2, 3, 4, 5, 6, 7\}$ exactly once is _______

gate2018-cse binary-heap numerical-answers combinatory

Answer

3.4.2 Binary Heap: GATE CSE 2020 | Question: 47 top ☛ https://gateoverflow.in/333184

Consider the array representation of a binary min-heap containing 1023 elements. The minimum number of comparisons required to find the maximum in the heap is ___________.

gate2020-cse numerical-answers binary-heap

Answer

3.4.3 Binary Heap: GATE CSE 2021 Set 2 | Question: 2 top ☛ https://gateoverflow.in/357538

Let H be a binary min-heap consisting of n elements implemented as an array. What is the worst case time complexity of an optimal algorithm to find the maximum element in H?

A. $\Theta(1)$
B. $\Theta(\log n)$
C. $\Theta(n)$
D. $\Theta(n \log n)$

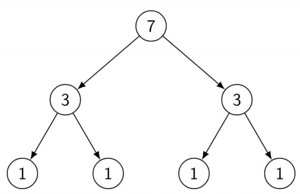
gate2021-cse-set2 data-structures heap binary-heap time-complexity

Answer

Answers: Binary Heap

3.4.1 Binary Heap: GATE CSE 2018 | Question: 46 top ☛ https://gateoverflow.in/204121

✔ Lets answer this question in an easier way :



Now do $\frac{7!}{7 \times 3 \times 3} = 80$

Here $7!$ because 7 items to be filled, Now 7 because root has only 7 nodes as its decedent including itself and only one can be the root. In same way we get 3 and 3 for the second level nodes and 1 and 1 for the third level.

👍 57 votes -- Prashant Singh (47.1k points)

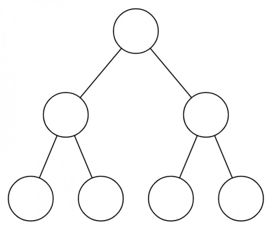
Ans: 80

Explanation:

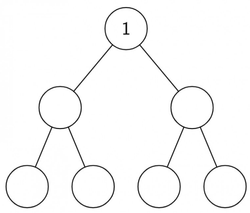
Number of min-heaps possible with keys $\{1, 2, 3, 4, 5, 6, 7\}$.

> A min-heap is a binary tree such that. - the data contained in each node is less than (or equal to) the data in that node's children. - the binary tree is complete.

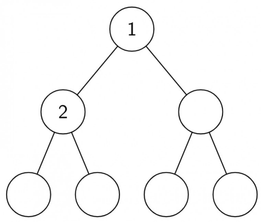
Since a binary heap is always a complete binary tree, so with 7 nodes, we can have 2 levels (root at level 0). It's structure will be like this:



Now because of min-heap property, every node's value must be smaller than all of it's children. So, this ensure that the minimum will always be at the root. $\therefore 1$ will be at the root.

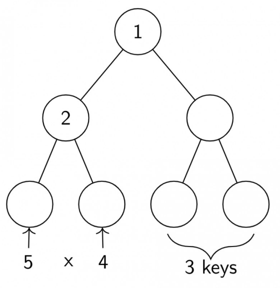


The second minimum element(i.e. 2) can be present at the first level only(root is at the zeroth level). So we have two options. Let's, for now, fix 2 at the left side.



We are now left with 5 more elements $\{3, 4, 5, 6, 7\}$. For the left subtree's two leaves, we have $5 * 4$ ways. i.e. first choosing one of the 5 nodes, then choosing one of the remaining 4 nodes.

Now 3 nodes left. Out of these 3, one will be the least among them, and that will definitely become the parent of the two remaining leaves(in the right subtree). Now with 2 nodes left, we can arrange them in 2 ways.



This gives $(5 * 4) * 2 = 40$ ways.

We can have the same number of ways, if we fixed 2 at the right subtree instead of left. So total ways:

$= 40 * 2$

$= \mathbf{80}$

282 votes -- Rishabh Gupta (12.5k points)

3.4.2 Binary Heap: GATE CSE 2020 | Question: 47 top https://gateoverflow.in/333184

✔ If a heap has 1023 elements, it'll contain 512 leaves and since it is a MIN-HEAP, the maximum will be present in the leaves. (Why? Assume it is not, then there will be some elements present under it and this will violate the min-heap property.)

We can visualise it this way. At each level starting with root level, number of elements are

$1 - 2 - 4 - 8 - 16 - 32 - 64 - 128 - 256 - 512$ (this is the last level, hence leaves)

So if we have n elements in an array, we know that to find the maximum it takes $n-1$ comparisons.

In this case, $n = 512$, so the answer should be 511.

Some other excellent questions on finding maximum-minimum:

- https://gateoverflow.in/1917/gate2014-1-39
- https://gateoverflow.in/27198/tifr2014-b-10
- https://gateoverflow.in/27194/tifr2014-b-9

References

19 votes -- goxul (5.2k points)

3.4.3 Binary Heap: GATE CSE 2021 Set 2 | Question: 2 top https://gateoverflow.in/357538

✔ In a min heap, maximum element is present in one of the leaf nodes.

If index of heap starts with 1, indices of leaves are $\lfloor n/2 \rfloor + 1, \lfloor n/2 \rfloor + 2, \lfloor n/2 \rfloor + 3 \ldots n$.

So, we have to perform linear search on at most $n/2 + 1$ elements to find the maximum element. (we cannot perform binary search since it is not guaranteed that leaves are in sorted order) and that multiplied by some constant c will be the time complexity of the optimal algorithm. (Here, c includes the cost of all operations which includes comparison, index shift etc. for a single maximum element compare)

In asymptotic terms, $c * (n/2 + 1)$ is $\Theta(n)$.

8 votes -- chirudeepnamini (3.2k points)

3.5 Binary Search Tree (31) top

3.5.1 Binary Search Tree: GATE CSE 1996 | Question: 2.14 top https://gateoverflow.in/2743

A binary search tree is generated by inserting in order the following integers:

$$50, 15, 62, 5, 20, 58, 91, 3, 8, 37, 60, 24$$

The number of nodes in the left subtree and right subtree of the root respectively is

A. $(4, 7)$
B. $(7, 4)$
C. $(8, 3)$
D. $(3, 8)$

gate1996 data-structures binary-search-tree normal

Answer

3.5.2 Binary Search Tree: GATE CSE 1996 | Question: 4 top https://gateoverflow.in/2756

A binary search tree is used to locate the number 43. Which of the following probe sequences are possible and which are not? Explain.

(a)	61	52	14	17	40	43
(b)	2	3	50	40	60	43
(c)	10	65	31	48	37	43
(d)	81	61	52	14	41	43
(e)	17	77	27	66	18	43

gate1996 data-structures binary-search-tree normal descriptive

Answer

3.5.3 Binary Search Tree: GATE CSE 1997 | Question: 4.5 top https://gateoverflow.in/2246

A binary search tree contains the value $1, 2, 3, 4, 5, 6, 7, 8$. The tree is traversed in pre-order and the values are printed out. Which of the following sequences is a valid output?

A. 5 3 1 2 4 7 8 6
B. 5 3 1 2 6 4 8 7
C. 5 3 2 4 1 6 7 8
D. 5 3 1 2 4 7 6 8

gate1997 data-structures binary-search-tree normal

Answer

3.5.4 Binary Search Tree: GATE CSE 2001 | Question: 14 top https://gateoverflow.in/755

A. Insert the following keys one by one into a binary search tree in the order specified.

$$15, 32, 20, 9, 3, 25, 12, 1$$

Show the final binary search tree after the insertions.

B. Draw the binary search tree after deleting 15 from it.

C. Complete the statements $S1$, $S2$ and $S3$ in the following function so that the function computes the depth of a binary tree rooted at t.

```
typedef struct tnode{
    int key;
    struct tnode *left, *right;
} *Tree;

int depth (Tree t)
{
    int x, y;
    if (t == NULL) return 0;
    x = depth (t -> left);
S1:     ___________;

S2:     if (x > y) return __________;

S3:     else return _______;

}
```

gate2001-cse data-structures binary-search-tree normal descriptive

Answer

3.5.5 Binary Search Tree: GATE CSE 2003 | Question: 19, ISRO2009-24 top https://gateoverflow.in/909

Suppose the numbers $7, 5, 1, 8, 3, 6, 0, 9, 4, 2$ are inserted in that order into an initially empty binary search tree. The binary search tree uses the usual ordering on natural numbers. What is the in-order traversal sequence of the resultant tree?

A. 7 5 1 0 3 2 4 6 8 9
B. 0 2 4 3 1 6 5 9 8 7
C. 0 1 2 3 4 5 6 7 8 9
D. 9 8 6 4 2 3 0 1 5 7

gate2003-cse binary-search-tree easy isro2009

Answer

3.5.6 Binary Search Tree: GATE CSE 2003 | Question: 6 top https://gateoverflow.in/897

Let $T(n)$ be the number of different binary search trees on n distinct elements.

Then $T(n) = \sum_{k=1}^{n} T(k-1)T(x)$, where x is

A. $n-k+1$
B. $n-k$
C. $n-k-1$
D. $n-k-2$

gate2003-cse normal binary-search-tree

Answer

3.5.7 Binary Search Tree: GATE CSE 2003 | Question: 63, ISRO2009-25 top https://gateoverflow.in/950

A data structure is required for storing a set of integers such that each of the following operations can be done in $O(\log n)$ time, where n is the number of elements in the set.

I. Deletion of the smallest element
II. Insertion of an element if it is not already present in the set

Which of the following data structures can be used for this purpose?

A. A heap can be used but not a balanced binary search tree
B. A balanced binary search tree can be used but not a heap
C. Both balanced binary search tree and heap can be used
D. Neither balanced search tree nor heap can be used

gate2003-cse data-structures easy isro2009 binary-search-tree

Answer

3.5.8 Binary Search Tree: GATE CSE 2004 | Question: 4, ISRO2009-26 top https://gateoverflow.in/1001

The following numbers are inserted into an empty binary search tree in the given order: $10, 1, 3, 5, 15, 12, 16$. What is the height of the binary search tree (the height is the maximum distance of a leaf node from the root)?

A. 2
B. 3
C. 4
D. 6

gate2004-cse data-structures binary-search-tree easy isro2009

Answer

3.5.9 Binary Search Tree: GATE CSE 2004 | Question: 85 top https://gateoverflow.in/1079

A program takes as input a balanced binary search tree with n leaf nodes and computes the value of a function $g(x)$ for each node x. If the cost of computing $g(x)$ is:

$$\min\left(\begin{array}{c}\text{number of leaf-nodes}\\\text{in left-subtree of } x\end{array}, \begin{array}{c}\text{number of leaf-nodes}\\\text{in right-subtree of } x\end{array}\right)$$

Then the worst-case time complexity of the program is?

A. $\Theta(n)$
B. $\Theta(n \log n)$

C. $\Theta(n^2)$
D. $\Theta(n^2 \log n)$

gate2004-cse binary-search-tree normal data-structures

Answer

3.5.10 Binary Search Tree: GATE CSE 2005 | Question: 33 top https://gateoverflow.in/1369

Postorder traversal of a given binary search tree, T produces the following sequence of keys

$10, 9, 23, 22, 27, 25, 15, 50, 95, 60, 40, 29$

Which one of the following sequences of keys can be the result of an in-order traversal of the tree T?

A. $9, 10, 15, 22, 23, 25, 27, 29, 40, 50, 60, 95$
B. $9, 10, 15, 22, 40, 50, 60, 95, 23, 25, 27, 29$
C. $29, 15, 9, 10, 25, 22, 23, 27, 40, 60, 50, 95$
D. $95, 50, 60, 40, 27, 23, 22, 25, 10, 9, 15, 29$

gate2005-cse data-structures binary-search-tree easy

Answer

3.5.11 Binary Search Tree: GATE CSE 2008 | Question: 46 top https://gateoverflow.in/458

You are given the postorder traversal, P, of a binary search tree on the n elements $1, 2, \ldots, n$. You have to determine the unique binary search tree that has P as its postorder traversal. What is the time complexity of the most efficient algorithm for doing this?

A. $\Theta(\log n)$
B. $\Theta(n)$
C. $\Theta(n \log n)$
D. None of the above, as the tree cannot be uniquely determined

gate2008-cse data-structures binary-search-tree normal

Answer

3.5.12 Binary Search Tree: GATE CSE 2012 | Question: 5 top https://gateoverflow.in/37

The worst case running time to search for an element in a balanced binary search tree with $n2^n$ elements is

A. $\Theta(n \log n)$
B. $\Theta(n2^n)$
C. $\Theta(n)$
D. $\Theta(\log n)$

gate2012-cse data-structures normal binary-search-tree

Answer

3.5.13 Binary Search Tree: GATE CSE 2013 | Question: 43 top https://gateoverflow.in/1554

The preorder traversal sequence of a binary search tree is $30, 20, 10, 15, 25, 23, 39, 35, 42$. Which one of the following is the postorder traversal sequence of the same tree?

A. $10, 20, 15, 23, 25, 35, 42, 39, 30$
B. $15, 10, 25, 23, 20, 42, 35, 39, 30$
C. $15, 20, 10, 23, 25, 42, 35, 39, 30$
D. $15, 10, 23, 25, 20, 35, 42, 39, 30$

gate2013-cse data-structures binary-search-tree normal

Answer

3.5.14 Binary Search Tree: GATE CSE 2013 | Question: 7 top https://gateoverflow.in/1416

Which one of the following is the tightest upper bound that represents the time complexity of inserting an object into a binary search tree of n nodes?

A. $O(1)$
B. $O(\log n)$
C. $O(n)$
D. $O(n \log n)$

gate2013-cse data-structures easy binary-search-tree

Answer

3.5.15 Binary Search Tree: GATE CSE 2014 Set 3 | Question: 39 top https://gateoverflow.in/2073

Suppose we have a balanced binary search tree T holding n numbers. We are given two numbers L and H and wish to sum up all the numbers in T that lie between L and H. Suppose there are m such numbers in T. If the tightest upper bound on the time to compute the sum is $O(n^a \log^b n + m^c \log^d n)$, the value of $a + 10b + 100c + 1000d$ is ______.

gate2014-cse-set3 data-structures binary-search-tree numerical-answers normal

First solve 3.5.22

Answer

3.5.16 Binary Search Tree: GATE CSE 2015 Set 1 | Question: 10 top https://gateoverflow.in/8129

Which of the following is/are correct in order traversal sequence(s) of binary search tree(s)?

I. $3, 5, 7, 8, 15, 19, 25$
II. $5, 8, 9, 12, 10, 15, 25$
III. $2, 7, 10, 8, 14, 16, 20$
IV. $4, 6, 7, 9, 18, 20, 25$

A. I and IV only
B. II and III only
C. II and IV only
D. II only

gate2015-cse-set1 data-structures binary-search-tree easy

Answer

3.5.17 Binary Search Tree: GATE CSE 2015 Set 1 | Question: 23 top https://gateoverflow.in/8221

What are the worst-case complexities of insertion and deletion of a key in a binary search tree?

A. $\Theta(\log n)$ for both insertion and deletion
B. $\Theta(n)$ for both insertion and deletion
C. $\Theta(n)$ for insertion and $\Theta(\log n)$ for deletion
D. $\Theta(\log n)$ for insertion and $\Theta(n)$ for deletion

gate2015-cse-set1 data-structures binary-search-tree easy

Answer

3.5.18 Binary Search Tree: GATE CSE 2015 Set 3 | Question: 13 top https://gateoverflow.in/8409

While inserting the elements $71, 65, 84, 69, 67, 83$ in an empty binary search tree (BST) in the sequence shown, the element in the lowest level is

A. 65
B. 67
C. 69
D. 83

gate2015-cse-set3 data-structures binary-search-tree easy

Answer

3.5.19 Binary Search Tree: GATE CSE 2016 Set 2 | Question: 40 top

https://gateoverflow.in/39586

The number of ways in which the numbers $1, 2, 3, 4, 5, 6, 7$ can be inserted in an empty binary search tree, such that the resulting tree has height 6, is _________.

Note: The height of a tree with a single node is 0.

gate2016-cse-set2 data-structures binary-search-tree normal numerical-answers

Answer

3.5.20 Binary Search Tree: GATE CSE 2017 Set 1 | Question: 6 top

https://gateoverflow.in/118286

Let T be a binary search tree with 15 nodes. The minimum and maximum possible heights of T are:

Note: The height of a tree with a single node is 0.

A. 4 and 15 respectively.
B. 3 and 14 respectively.
C. 4 and 14 respectively.
D. 3 and 15 respectively.

gate2017-cse-set1 data-structures binary-search-tree easy

Answer

3.5.21 Binary Search Tree: GATE CSE 2017 Set 2 | Question: 36 top

https://gateoverflow.in/118378

The pre-order traversal of a binary search tree is given by $12, 8, 6, 2, 7, 9, 10, 16, 15, 19, 17, 20$. Then the post-order traversal of this tree is

A. $2, 6, 7, 8, 9, 10, 12, 15, 16, 17, 19, 20$
B. $2, 7, 6, 10, 9, 8, 15, 17, 20, 19, 16, 12$
C. $7, 2, 6, 8, 9, 10, 20, 17, 19, 15, 16, 12$
D. $7, 6, 2, 10, 9, 8, 15, 16, 17, 20, 19, 12$

gate2017-cse-set2 data-structures binary-search-tree

Answer

3.5.22 Binary Search Tree: GATE CSE 2020 | Question: 41 top

https://gateoverflow.in/333190

In a balanced binary search tree with n elements, what is the worst case time complexity of reporting all elements in range $[a, b]$? Assume that the number of reported elements is k.

A. $\Theta(\log n)$
B. $\Theta(\log n + k)$
C. $\Theta(k \log n)$
D. $\Theta(n \log k)$

gate2020-cse data-structures binary-search-tree

Answer

3.5.23 Binary Search Tree: GATE CSE 2020 | Question: 5 top

https://gateoverflow.in/333226

The preorder traversal of a binary search tree is $15, 10, 12, 11, 20, 18, 16, 19$. Which one of the following is the postorder traversal of the tree?

A. $10, 11, 12, 15, 16, 18, 19, 20$
B. $11, 12, 10, 16, 19, 18, 20, 15$
C. $20, 19, 18, 16, 15, 12, 11, 10$
D. $19, 16, 18, 20, 11, 12, 10, 15$

gate2020-cse binary-search-tree

Answer

3.5.24 Binary Search Tree: GATE CSE 2021 Set 1 | Question: 10 top https://gateoverflow.in/357442

A binary search tree T contains n distinct elements. What is the time complexity of picking an element in T that is smaller than the maximum element in T?

A. $\Theta(n \log n)$
B. $\Theta(n)$
C. $\Theta(\log n)$
D. $\Theta(1)$

gate2021-cse-set1 data-structures binary-search-tree time-complexity

Answer

3.5.25 Binary Search Tree: GATE IT 2005 | Question: 12 top https://gateoverflow.in/3757

The numbers $1, 2, \dots . n$ are inserted in a binary search tree in some order. In the resulting tree, the right subtree of the root contains p nodes. The first number to be inserted in the tree must be

A. p
B. $p+1$
C. $n-p$
D. $n-p+1$

gate2005-it data-structures normal binary-search-tree

Answer

3.5.26 Binary Search Tree: GATE IT 2005 | Question: 55 top https://gateoverflow.in/3816

A binary search tree contains the numbers $1, 2, 3, 4, 5, 6, 7, 8$. When the tree is traversed in pre-order and the values in each node printed out, the sequence of values obtained is $5, 3, 1, 2, 4, 6, 8, 7$. If the tree is traversed in post-order, the sequence obtained would be

A. $8, 7, 6, 5, 4, 3, 2, 1$
B. $1, 2, 3, 4, 8, 7, 6, 5$
C. $2, 1, 4, 3, 6, 7, 8, 5$
D. $2, 1, 4, 3, 7, 8, 6, 5$

gate2005-it data-structures binary-search-tree normal

Answer

3.5.27 Binary Search Tree: GATE IT 2006 | Question: 45 top https://gateoverflow.in/3588

Suppose that we have numbers between 1 and 100 in a binary search tree and want to search for the number 55. Which of the following sequences CANNOT be the sequence of nodes examined?

A. $\{10, 75, 64, 43, 60, 57, 55\}$
B. $\{90, 12, 68, 34, 62, 45, 55\}$
C. $\{9, 85, 47, 68, 43, 57, 55\}$
D. $\{79, 14, 72, 56, 16, 53, 55\}$

gate2006-it data-structures binary-search-tree normal

Answer

3.5.28 Binary Search Tree: GATE IT 2007 | Question: 29 top https://gateoverflow.in/3462

When searching for the key value 60 in a binary search tree, nodes containing the key values $10, 20, 40, 50, 70, 80, 90$ are traversed, not necessarily in the order given. How many different orders are possible in which these key values can occur on the search path from the root to the node containing the value 60?

A. 35
B. 64

C. 128
D. 5040

gate2007-it data-structures binary-search-tree normal

Answer

3.5.29 Binary Search Tree: GATE IT 2008 | Question: 71 top https://gateoverflow.in/3385

A Binary Search Tree (BST) stores values in the range 37 to 573. Consider the following sequence of keys.

I. $81, 537, 102, 439, 285, 376, 305$
II. $52, 97, 121, 195, 242, 381, 472$
III. $142, 248, 520, 386, 345, 270, 307$
IV. $550, 149, 507, 395, 463, 402, 270$

Suppose the BST has been unsuccessfully searched for key 273. Which all of the above sequences list nodes in the order in which we could have encountered them in the search?

A. II and III only
B. I and III only
C. III and IV only
D. III only

gate2008-it data-structures binary-search-tree normal

Answer

3.5.30 Binary Search Tree: GATE IT 2008 | Question: 72 top https://gateoverflow.in/3386

A Binary Search Tree (BST) stores values in the range 37 to 573. Consider the following sequence of keys.

I. $81, 537, 102, 439, 285, 376, 305$
II. $52, 97, 121, 195, 242, 381, 472$
III. $142, 248, 520, 386, 345, 270, 307$
IV. $550, 149, 507, 395, 463, 402, 270$

Which of the following statements is TRUE?

A. I, II and IV are inorder sequences of three different BSTs
B. I is a preorder sequence of some BST with 439 as the root
C. II is an inorder sequence of some BST where 121 is the root and 52 is a leaf
D. IV is a postorder sequence of some BST with 149 as the root

gate2008-it data-structures binary-search-tree easy

Answer

3.5.31 Binary Search Tree: GATE IT 2008 | Question: 73 top https://gateoverflow.in/3387

How many distinct BSTs can be constructed with 3 distinct keys?

A. 4
B. 5
C. 6
D. 9

gate2008-it data-structures binary-search-tree normal

Answer

Answers: Binary Search Tree

3.5.1 Binary Search Tree: GATE CSE 1996 | Question: 2.14 top https://gateoverflow.in/2743

✔ Correct Option: B

Root will be 50. now insert one by one, greater to 50 in the right sub tree, lesser in left sub tree.

Or you can simply count the number looking at the i/p. less than 50 are 7. more than 50 are 4.

34 votes -- Gate Keeda (15.9k points)

3.5.2 Binary Search Tree: GATE CSE 1996 | Question: 4 top https://gateoverflow.in/2756

✔ **(b)** and **(e)** are not possible.

rest all i/p's will have binary trees with only one child. but (b) and (e) will have two childs at a point. therefore the probe sequence will not be possible.

For better clarification, make BST's for the given i/p's and probe for 43.

38 votes -- Gate Keeda (15.9k points)

3.5.3 Binary Search Tree: GATE CSE 1997 | Question: 4.5 top https://gateoverflow.in/2246

✔ Note: PreOrder means Root-Left-Right Traversal of tree.

By Option Elimination:-

B. False. Because 5 is root so in preorder sequence first 5 elements must contain 1 to 5 But 6 comes here. So false.

So, now common things in option A,C,D is $5, 3$ means 5 root then LHS subtree root is 3 . now 3 s LHS is $1, 2$ so they should come first rather than 4. So option C is False.

Now Check for **Option A,D.**

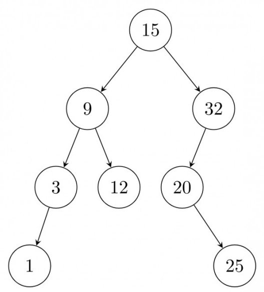
Root 5's RHS is $6, 7, 8$ now as per Option A,D 7 is Root so 6 should be Left and 8 should be Right so pre order for Right sub tree is $7, 6, 8$ (Root-L-R). This satisfies option **D**.

Correct Answer: D

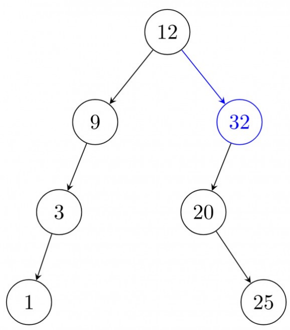
45 votes -- Rajesh Pradhan (18.9k points)

3.5.4 Binary Search Tree: GATE CSE 2001 | Question: 14 top https://gateoverflow.in/755

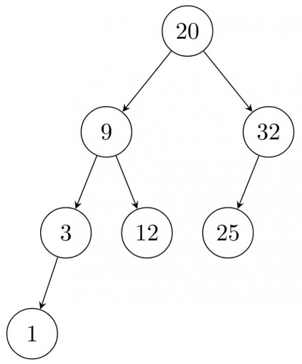
✔ Binary search tree will be:



After delete of root: use inorder predecessor



Or After delete of root: use inorder successor




```
typedef struct tnode{
    int key;
    struct tnode *left, *right;
} *Tree;

int depth (Tree t)
{
    int x, y;
    if (t == NULL) return 0;
    x = depth (t -> left);
S1: y = depth (t -> right);

S2:     if (x > y) return (1+x);one for root

S3:     else return (1+y); one for root

}
```

28 votes -- Anu007 (14.4k points)

3.5.5 Binary Search Tree: GATE CSE 2003 | Question: 19, ISRO2009-24 top https://gateoverflow.in/909

✔ In-order traversal returns the elements in ascending (smallest to largest) order.

Therefore, correct option **C**

27 votes -- Gate_15_isHere (459 points)

3.5.6 Binary Search Tree: GATE CSE 2003 | Question: 6 top https://gateoverflow.in/897

✔ The summation is for each node, if that node happens to be the root. When a node is root, it will have $(k-1)$ nodes on the left sub tree (k being any number) and correspondingly $(n-k)$ elements on the right sub tree. So, we can write recurrence $T(k-1) * T(n-k)$ for the number of distinct binary search trees, as the numbers on left and right sub trees form BSTs independent of each other and only a difference in one of the sub trees produces a difference in the tree. Hence, answer is **B**.

Knowing the direct formula can also help in getting the answer but is not recommended.

https://gatecse.in/number-of-binary-trees-possible-with-n-nodes/

58 votes -- Arjun Suresh (330k points)

3.5.7 Binary Search Tree: GATE CSE 2003 | Question: 63, ISRO2009-25 top https://gateoverflow.in/950

✔ Balanced search tree have height $\log n$

Deletion of smallest element will take $O(\log n)$ time

Finding a element is present/not and doing insertion: $O(\log n)$

Heap(MIN) is also an almost complete binary tree have height $\log n$

Deletion of smallest element will take $O(\log n)$ time (root element removal, replace with last element +balancing)

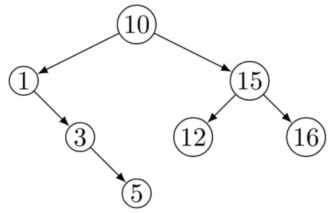
Finding a element is present/not and insertion: Finding only takes $O(n)$, insertion then balancing take $O(\log n)$. So, total $O(n) + O(\log n) = O(n)$.

Answer is **B.**

(even if its maxheap our ans does not change only time for deletion of min will increase $O(n)$)

73 votes -- Anurag Semwal (6.7k points)

3.5.8 Binary Search Tree: GATE CSE 2004 | Question: 4, ISRO2009-26 top https://gateoverflow.in/1001



Height is 3

Correct Answer: B

33 votes -- Prashant Singh (47.1k points)

3.5.9 Binary Search Tree: GATE CSE 2004 | Question: 85 top https://gateoverflow.in/1079

B. At the root node (first level) the cost would be $\Theta\left(\frac{n}{2}\right)$ as the tree is **balanced**.

At next level, we have 2 nodes and for each of them cost of computing $g(x)$ will be $\Theta\left(\frac{n}{4}\right)$. So, total cost at second level $= \Theta\left(\frac{n}{2}\right)$. Similarly at **each level** (total cost per level and not the cost per node in a level) the cost would be $\Theta\left(\frac{n}{2}\right)$ and so for $\log n$ levels it would be $\Theta(n \log n)$.

PS: Even if we change $\min$ to $\max$ in the defintion of $g(x)$ we get the same answer.

85 votes -- Shaun Patel (6.1k points)

3.5.10 Binary Search Tree: GATE CSE 2005 | Question: 33 top https://gateoverflow.in/1369

In order traversal of b binary search tree returns the element in sorted order - ascending (inorder is left parent then right. in a bst left is less than parent and right is greater than parent). In this option A is the only sorted list. hence it is the only possibility.

31 votes -- Sankaranarayanan P.N (8.5k points)

3.5.11 Binary Search Tree: GATE CSE 2008 | Question: 46 top https://gateoverflow.in/458

Correct Answer: B

Last element in post order is the root of tree- find this element in inorder- $\log n$ time.
Now as in quick sort consider this as pivot and split the post order array into 2- possible because all elements smaller than pivot goes to left and all elements larger than pivot goes to right and suppose we have x elements smaller than pivot, these elements will be same in both inorder as well as postorder (order may change). We already got the root, now left child is the left split and right child is the right split.

So, doing this recursively gives time complexity of this approach as

$$T(n) = T(k) + T(n - k - 1) + \log n$$

Solving would give $T(n) = O(n \log n)$ in worst case, by putting $k = 0$ and shown at bottom.

But searching for an element in the inorder traversal of given BST can be done in $O(1)$ because we have n elements from $1..n$ so there is no need to search for an element- if last element in post order is say 5 we take it as root and since 4 elements $(1..4)$ are smaller we split the post order array in to two- (first 4 elements), (6th element onward) and solve recursively. Time complexity for this would be

$$T(n) = T(k) + T(n - k - 1) + O(1)$$

which gives $T(n) = O(n)$.

Since we know that all elements must be traversed at least once, $T(n) = \Omega(n)$ also and so

$$T(n) = \Theta(n).$$

The following code is doing this.

```
//Install graphviz (sudo apt-get install graphviz on Ubuntu) to view output tree
#include<stdio.h>
#include<stdlib.h>
struct tree
{
    struct tree* left;
    struct tree* right;
    int x;
};
struct tree* makenode(int x)
{
    struct tree * root =  malloc(sizeof(struct tree));
    root -> x = x;
    root -> left = root -> right = NULL;
    return root;
}

struct tree* makeBST(int *post, int start, int n, int inorder){
    if(n <= 0)
            return NULL;
    int pivot = post[start + n -1];
    struct tree * root = makenode(pivot);
    root -> left = makeBST(post, start, pivot-1 - inorder, inorder );
    root -> right = makeBST(post, pivot - inorder - 1, n - (pivot - inorder), pivot);
    return root;
}
void preorder(struct tree* node)
{
    if(node == NULL)
        return;
    printf("%d ", node->x);
    preorder(node->left);
    preorder(node->right);
}
void printdot(struct tree* node, FILE * f)
{
    if(node == NULL)
        return;
    if(node-> left != NULL)
    {
        fprintf(f, "%d -- %d;\n", node->x, node->left->x);
    }
    if(node-> right != NULL)
    {
        fprintf(f, "%d -- %d;\n", node->x, node->right->x);
    }
    printdot(node->left, f);
    printdot(node->right, f);
}

int main(){
    int i, n, *a;
    printf("Enter n: ");
    scanf("%d", &n);
    a = malloc(n * sizeof(int));
    printf ("Enter post order traversal: ");
    for(i = 0; i < n; i++)
    {
        scanf("%d", &a[i]);
    }
    struct tree * tree = makeBST(a, 0, n, 0);
    printf("Pre order traversal is : ");
    preorder(tree);
    printf("\n");
    FILE * f = fopen("tree.dot", "w");
    fprintf(f, "graph tree { \n");
    printdot(tree, f);
    fprintf(f, "}\n");
    fclose(f);

    #if defined(__linux__) || (defined(__APPLE__) && defined(__MACH__) || (defined (__gnu_linux__)) )
        system("dot -Tpng tree.dot -o output.png; eog output.png");
    #endif

}
```

$$T(n) = T(k) + T(n-k-1) + \log n$$

Solving would give $T(n) = O(n \log n)$, by putting $k = 0$,
$T(n) = T(0) + T(n-1) + \log n, \implies T(n) = O(1) + \log n + \log(n-1) + \log(n-2) + \cdots + \log 1 \implies T(n) = n$
([Stirling's Approximation](#))

PS: Even for a general BST, given a post order traversal, we can construct the BST in $O(n)$ more stricter than $O(n \log n)$ derived above and this matches $\Omega(n)$ and hence we do have an $\Theta(n)$ algorithm. This algorithm can be seen at below link
https://www.geeksforgeeks.org/construct-a-binary-search-tree-from-given-postorder/

References

96 votes -- Arjun Suresh (330k points)

3.5.12 Binary Search Tree: GATE CSE 2012 | Question: 5 top https://gateoverflow.in/37

✔ Binary search takes $\Theta(\log n)$ for n elements in the worst case. So, with $(n2^n)$ elements, the worst case time will be

$\Theta(\log(n2^n))$

$= \Theta(\log n + \log 2^n)$

$= \Theta(\log n + n)$

$= \Theta(n)$

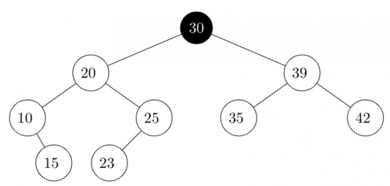
Correct Answer: C

79 votes -- Arjun Suresh (330k points)

3.5.13 Binary Search Tree: GATE CSE 2013 | Question: 43 top https://gateoverflow.in/1554

✔ Since it is a binary search tree, its inorder traversal produces a sorted sequence i.e.
$10, 15, 20, 23, 25, 30, 35, 39, 42$.

Now given inorder and preorder traversals, we get following tree :



From this, we can give postorder traversal sequence as $15, 10, 23, 25, 20, 35, 42, 39, 30$ i.e., **option (D)**.

35 votes -- Happy Mittal (8.2k points)

3.5.14 Binary Search Tree: GATE CSE 2013 | Question: 7 top https://gateoverflow.in/1416

✔ Option (**C**) is True .
Suppose that we need to insert a node z such that $k = key[z]$. Using binary search we find a nil such that replacing it by z does not break the BST-property

BST-Insert$(\mathbf{x, z, k})$

: if $x = nil$ then return "Error"
: $y \leftarrow x$
: while true do {
: if $key[y] < k$
: then $z \leftarrow left[y]$
: else $z \leftarrow right[y]$

7. : if $z = nil$ break
8. : }
9. : if $key[y] > k$ then $left[y] \leftarrow z$
10. : else $right[p[y]] \leftarrow z$

Time Complexity Analysis :

1. Best $Case = O(1)$, When it is smallest/greatest element and BST contains only all greater/smaller element than inserting element respectively.
2. Avg $Case = O(\log n)$, When it belongs between some elements .
3. Worst $Case = O(n)$, When it is smallest/greatest element and BST contains only all smaller/greater element than inserting element respectively.

👍 42 votes -- Bhagirathi Nayak (11.7k points)

3.5.15 Binary Search Tree: GATE CSE 2014 Set 3 | Question: 39 top ☛ https://gateoverflow.in/2073

✔ In worst case for finding L and H it will take $O(\log n)$ time as the given tree is **balanced** binary search tree. Now there are m elements between L and H. So to traverse m element it will take $O(m)$ time (traversal algorithm given below). So, total

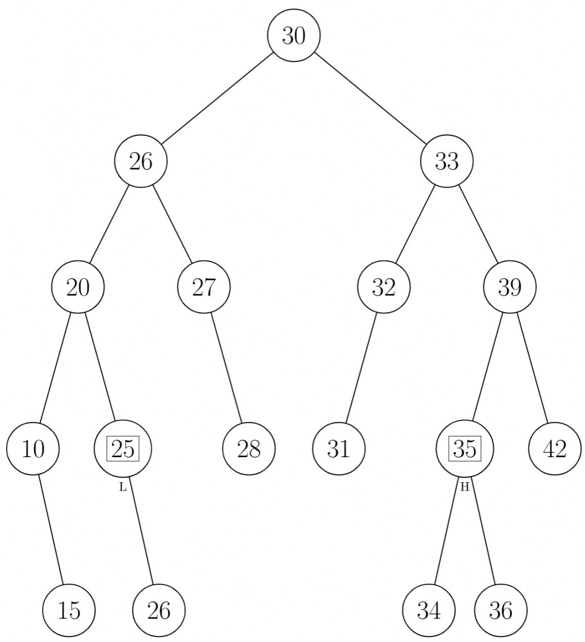
$O(m + \log n) \implies a = 0, b = 1, c = 1, d = 0$
$\therefore 0 + (10 \times 1) + (100 \times 1) + (1000 \times 0) = 110.$

To find all the numbers from L to H we can do an inorder traversal from root and discard all elements before L and after H. But this has $O(n)$ time complexity. So, we can do a modification to inorder traversal and combine with binary search as follows:

1. Find L using binary search and keep all nodes encountered in the search in a stack.
2. After finding L add it to stack as well and initialize $sum = 0$.
3. Now, for all nodes in stack, do an inorder traversal starting from their right node and adding the node value to sum. If H is found, stop the algorithm.

👍 125 votes -- Kalpish Singhal (1.6k points)

Here is an example 😊



$L = 25, H = 35$
Inorder: $10, 15, 20, \underset{L}{\boxed{25}}, 26, 27, 28, 30, 31, 32, 33, 34, \underset{H}{\boxed{35}}, 36, 39, 42$

1. Find L and $H \rightarrow O(\log n)$ time.

2. Traverse 'm' elements between 'L' and 'H' $\rightarrow O(m)$ [Traversal algorithm]

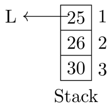
Total $\rightarrow O(m + \log n)$

$a = 0, b = 1, c = 1, d = 0$

$10 + 100 = 110$ Answer

Traversal Algorithm:

1. Find L using Binary search and keep $H > node > L$ encountered in the search in a stack.



2. After finding L, add it to stack as well & initialize $sum = 0$
3. Now, for all nodes in the stack, do an inorder traversal starting from their right node and adding the node value to sum. If it is found than stop the algorithm.

- Node (1)
  - No Right child; Sum = Sum + Node value $= 0 + 25 = 25.$
- Node (2)
  - Inorder Traversal from Right Child $\rightarrow 24, 28$
  - Sum = sum + inorder Traversal + Node Value $= (25 + 27 + 28) + 26$
- Node (3)
  - Inorder Traversal From Right Child $\rightarrow 31, 32, 33, 34, \overset{H}{\mathbf{35}} \rightarrow$ Stop Inorder Traversal
  - Sum = Sum + Inorder Traversal + Node value
    - $= 25 + 26 + 24 + 28 + (31 + 32 + 33 + 34 + 35) + 30$
    - $= 25 + 26 + 24 + 28 + 30 + 31 + 32 + 33 + 34 + 35$ Answer

---

EDIT :

1. In Step 1, we need to find only L and not H.
2. In Traversal Algo: Find L using Binary Search and Keep, **L< Nodes< H**, encountered in the search in the stack.

79 votes -- Vidhi Sethi (8.3k points)

3.5.16 Binary Search Tree: GATE CSE 2015 Set 1 | Question: 10 top https://gateoverflow.in/8129

✔ In order traversal of key are always in ascending order.

So, here I & IV th sequence are in ascending order so Option **A** is Answer.

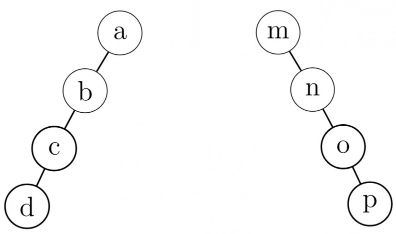
23 votes -- Rajesh Pradhan (18.9k points)

3.5.17 Binary Search Tree: GATE CSE 2015 Set 1 | Question: 23 top https://gateoverflow.in/8221

✔ Option **B**, both happens when the BST is skewed.

What is a **Skewed Tree?**

A binary tree which is dominated solely by left child nodes or right child nodes is called a **skewed binary tree** or more specifically **left skewed binary tree** or **right skewed binary tree** respectively.



Left Skewed Right Skewed

39 votes -- GATERush (917 points)

3.5.18 Binary Search Tree: GATE CSE 2015 Set 3 | Question: 13 top https://gateoverflow.in/8409

✔ On last level 67, hence option B is True.

33 votes -- Bran Stark (339 points)

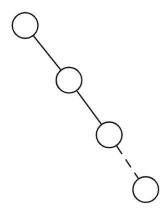
3.5.19 Binary Search Tree: GATE CSE 2016 Set 2 | Question: 40 top https://gateoverflow.in/39586

✔ We need to fill 7 levels with 7 elements. So, at each level we have exactly 2 possible options like 1 and 7 for root- one corresponding to making it left skewed and other right skewed. And this is the same for all levels up to 6 giving $2^6 = 64$ possible ways.

141 votes -- Arjun Suresh (330k points)

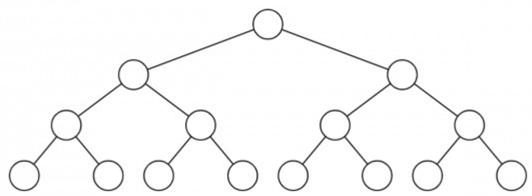
3.5.20 Binary Search Tree: GATE CSE 2017 Set 1 | Question: 6 top https://gateoverflow.in/118286

✔ The maximum height of a binary search tree will be when the tree is fully skewed



Maximum height $= n-1 = 15{-}1 = 14$

The minimum height of a binary search tree will be when the tree is full.



Minimum height $= \log_2(n+1){-}1 = \log_2(15+1){-}1 = \log_2(16){-}1 = 4-1 = 3$

Correct Answer: B

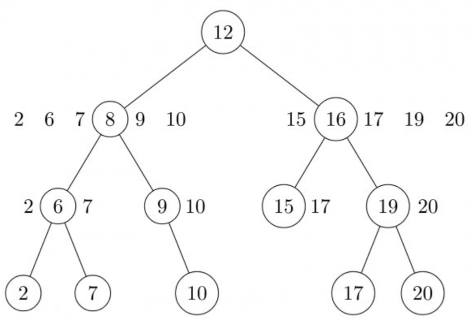
17 votes -- Pankaj Kumar (7.8k points)

3.5.21 Binary Search Tree: GATE CSE 2017 Set 2 | Question: 36 top https://gateoverflow.in/118378

✔ Preoder: 12 8 6 2 7 9 10 16 15 19 17 20

Inorder of BST must be sorted in increasing order!

Inorder: $\underbrace{\overbrace{2\quad 6\quad 7}^{\text{left}}\quad \overset{\text{root}}{8}\quad \overbrace{9\quad 10}^{\text{right}}}_{\text{left}}\quad \overset{\text{root}}{12}\quad \underbrace{15\quad 16\quad 17\quad 19\quad 20}_{\text{right}}$



So, Postorder: 2 7 6 10 9 8 15 17 20 19 16 12.

Answer is **B.**

👍 23 votes -- 2018 (5.5k points)

3.5.22 Binary Search Tree: GATE CSE 2020 | Question: 41 top ☛ https://gateoverflow.in/333190

✔ First, you'll have to check if the elements a and b are present in the BST or not. Since the BST is balanced, it will take $\Theta(\log_2(n))$ time for that. Traversing the k elements would take additional $\Theta(k)$ time.

Hence $\Theta(\log_2(n) + k)$

👍 17 votes -- Debasish Das (1.5k points)

3.5.23 Binary Search Tree: GATE CSE 2020 | Question: 5 top ☛ https://gateoverflow.in/333226

✔ Preorder traversal of BST : $15, 10, 12, 11, 20, 18, 16, 19$

In Pre-order, the first node is ROOT. So root is 15.

In Post-order, the last node is ROOT. So in the Post-order sequence, 15 must be at last.

In Pre-order, the second node is the left child of ROOT, if it is less than the root.

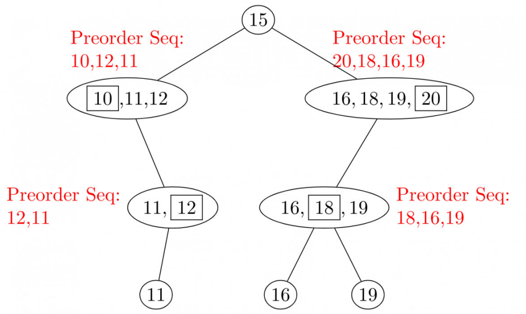
Sequence: $10, 12, 11$ **belongs to the left sub-tree of ROOT.**

$10, 12, 11$ is a Preorder of BST -- repetitively apply the steps.

In the Pre-order, the nodes which are greater than ROOT are on the right side of ROOT.

Sequence: $20, 18, 16, 19$ **belongs to the right sub-tree of ROOT.**

$20, 18, 16, 19$ is a Preorder of BST -- repetitively apply the steps.



Finally we will get $11, 12, 10, 16, 19, 18, 20, 15$ as Postorder.

👍 9 votes -- Shaik Masthan (50.4k points)

3.5.24 Binary Search Tree: GATE CSE 2021 Set 1 | Question: 10 top ☛ https://gateoverflow.in/357442

✔ If our data structure contains n distinct elements then :

In all the standard data structures that we know/study about, If we want to pick/find an element which is Not maximum (smaller than maximum) then time complexity will be $\Theta(1)$ because we only need to compare any two elements. Take any two elements that you can access in constant time, compare them and return smaller of those two elements.

PS :

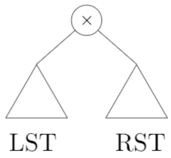
By "standard data structures that we know/study about" I mean the following :

Binary tree, Binary search tree, AVL tree, sorted or unsorted array, linked lists, arrays, stacks, queues, hash tables, heaps etc.

👍 7 votes -- Deepak Poonia (23.3k points)

3.5.25 Binary Search Tree: GATE IT 2005 | Question: 12 top ☛ https://gateoverflow.in/3757

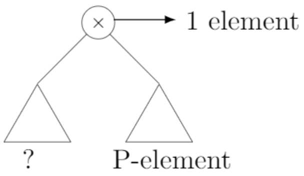
✔ Option is **C.**



P: # elements in RST

$\Rightarrow$ Depending on X, some number will go LST $\notin$ RST

$\Rightarrow$



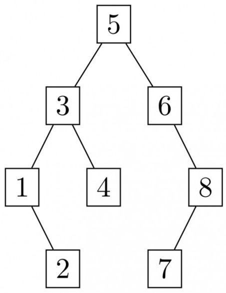
Remaining elements for LST: $n-(p+1)$

$$\underbrace{1,2,\ldots,n-(p+1)}_{\text{LST}n-(p+1)\text{elements}} \quad \underbrace{n-p}_{\text{Root1element}} \quad \underbrace{(n-p)+1,\ldots,n}_{\text{RST}p\text{elements}}$$

58 votes -- Akhil Nadh PC (16.5k points)

3.5.26 Binary Search Tree: GATE IT 2005 | Question: 55 top https://gateoverflow.in/3816

✔



The answer is D.

30 votes -- Gate Keeda (15.9k points)

3.5.27 Binary Search Tree: GATE IT 2006 | Question: 45 top https://gateoverflow.in/3588

✔ In option **C** search sequence progress in $\ldots 47, 68, 43, \ldots$

At 47 we see that search key 55 is greater and it will be on right side of 47. so in further comparison a value less than 47 will not come

Hence, option **C** is wrong.

38 votes -- Sankaranarayanan P.N (8.5k points)

3.5.28 Binary Search Tree: GATE IT 2007 | Question: 29 top https://gateoverflow.in/3462

✔ $10, 20, 40, 50, 70, 80, 90$

In BST search we if we go from say 10 to 40 while searching for 60, we will never encounter 20. So, $10, 20, 40$ and 50 visited, means they are visited in order. Similarly, $90, 80$ and 70 are visited in order. So, our required answer will be

$$\frac{\text{No. of possible permutations of 7 numbers}}{\text{No. of possible permutations of numbers smaller than } 60 \times \text{No. of possible permutations of numbers larger than } 60}$$

(Since only one permutation is valid for both the smaller set of numbers as well as larger set of numbers)

$= \frac{7!}{4!3!}$

$= 35$

Correct Answer: A

👍 169 votes -- Arjun Suresh (330k points)

Question is similar to this question : https://gateoverflow.in/1275/gate2007_84-85

We will convert Moves to Text.
It is given that During Search we have **Traversed** these nodes

$$\{10, 20, 40, 50, 70, 80, 90\}$$

as it can be seen that Red ones are bigger than 60 and blue ones are smaller than 60.
Path to node 60 has involved those nodes. So, one of the possible solution to the problem is

$$\{L, L, L, L, S, S, S\}.$$

Any other solution will contains these moves only because at a time on a node we can get directions as S(meaning 60 is smaller) or L(meaning 60 is larger) on comparison and since it is given that those nodes were encountered it means directions were picked from that set.

Hence, total number of possible solutions = all Permutations of that set, which is given by $\frac{7!}{4! \times 3!} = 35$

answer = **option A**

References

👍 42 votes -- Amar Vashishth (25.2k points)

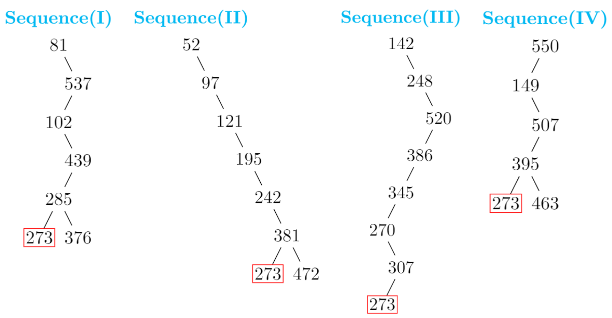
3.5.29 Binary Search Tree: GATE IT 2008 | Question: 71 top

https://gateoverflow.in/3385

✔ The option is **D.**

> Which all of the above sequences list nodes in the order in which we could have encountered them in the search?

The question goes like this. IF there had been $\mathbf{273}$ in the actual sequence then which of the following search sequences would have been successful.



In sequence 1 no need to go from 285 to 376 as 273 is less than 285.

In sequence 2 no need to go from 381 to 472 as 273 is less than 381.

In sequence 4 no need to go from 395 to 463 as 273 is less than 395.

In sequence 3 number 273 might have been to the left of 307 and search would have been successful.

Hence, **Option D**

👍 54 votes -- Akhil Nadh PC (16.5k points)

Answer: D

I. no need to go from 285 to 376 as 273 is less than 285.

II. no need to go from 381 to 472 as 273 is less than 381.

IV. no need to go from 395 to 463 as 273 is less than 395.

28 votes -- Rajarshi Sarkar (27.8k points)

3.5.30 Binary Search Tree: GATE IT 2008 | Question: 72 top https://gateoverflow.in/3386

A. Incorrect because I & IV are not in ascending order.(Inoder sequence of BST is in increasing order)

B. I is a preorder sequence of some BST with 439 as the root . False because if 439 is root, it should be first element in preorder.

C. This is correct.

D. IV is a postorder sequence of some BST with 149 as the root, False because if 149 is root, it should be last element in postorder

32 votes -- Akash Kanase (36k points)

3.5.31 Binary Search Tree: GATE IT 2008 | Question: 73 top https://gateoverflow.in/3387

For number of distinct BSTs with n nodes we apply the formula

$$\frac{C(2n,n)}{n+1}$$

$n = 3$ here, so $C(6,3) = 20$

So, $\frac{C(2n,n)}{n+1} = 20/4 = 5$

Answer is $\mathbf{5}$

REF :- https://gatecse.in/number-of-binary-trees-possible-with-n-nodes/

Correct Answer: B

References

41 votes -- Abhimanyu Kumar (161 points)

3.6 Binary Tree (50) top

3.6.1 Binary Tree: GATE CSE 1987 | Question: 2c top https://gateoverflow.in/80579

State whether the following statements are TRUE or FALSE:

It is possible to construct a binary tree uniquely whose pre-order and post-order traversals are given?

gate1987 binary-tree data-structures normal true-false

Answer

3.6.2 Binary Tree: GATE CSE 1987 | Question: 2g top https://gateoverflow.in/80588

State whether the following statements are TRUE or FALSE:

If the number of leaves in a tree is not a power of 2, then the tree is not a binary tree.

gate1987 data-structures binary-tree true-false

Answer

3.6.3 Binary Tree: GATE CSE 1987 | Question: 7b top

https://gateoverflow.in/82427

Construct a binary tree whose preorder traversal is

- $K\ L\ N\ M\ P\ R\ Q\ S\ T$

and inorder traversal is

- $N\ L\ K\ P\ R\ M\ S\ Q\ T$

gate1987 data-structures binary-tree descriptive

Answer

3.6.4 Binary Tree: GATE CSE 1988 | Question: 7i top

https://gateoverflow.in/94366

Define the height of a binary tree or subtree and also define a height-balanced (AVL) tree.

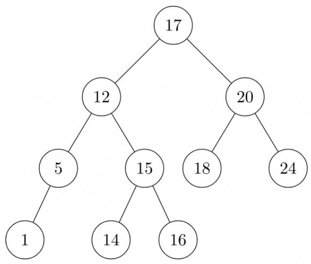
gate1988 normal descriptive data-structures binary-tree

Answer

3.6.5 Binary Tree: GATE CSE 1988 | Question: 7ii top

https://gateoverflow.in/94367

Mark the balance factor of each on the tree given on the below figure and state whether it is height-balanced.



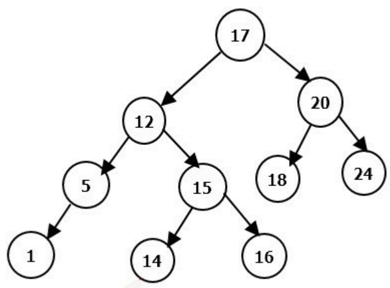
gate1988 data-structures normal descriptive binary-tree

Answer

3.6.6 Binary Tree: GATE CSE 1988 | Question: 7iii top

https://gateoverflow.in/94368

Consider the tree given in the below figure, insert 13 and show the new balance factors that would arise if the tree is not rebalanced. Finally, carry out the required rebalancing of the tree and show the new tree with the balance factors on each mode.



gate1988 normal descriptive data-structures binary-tree

Answer

3.6.7 Binary Tree: GATE CSE 1990 | Question: 3-iv top

https://gateoverflow.in/84828

The total external path length, EPL, of a binary tree with n external nodes is, $\text{EPL} = \sum_w I_w$, where I_w is the path length of external node w),

A. $\leq n^2$ always.
B. $\geq n \log_2 n$ always.

All external nodes are leaf nodes and the internal nodes are non-leaf nodes.

C. Equal to n^2 always.
D. $O(n)$ for some special trees.

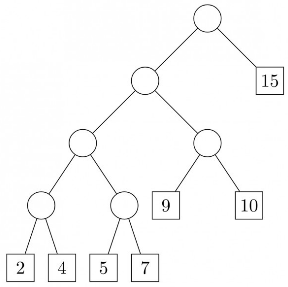
gate1990 normal data-structures binary-tree multiple-selects

Answer

3.6.8 Binary Tree: GATE CSE 1991 | Question: 01,viii top

https://gateoverflow.in/506

The weighted external path length of the binary tree in figure is ______



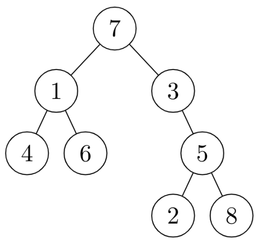
gate1991 binary-tree data-structures normal numerical-answers

Answer

3.6.9 Binary Tree: GATE CSE 1991 | Question: 1,ix top

https://gateoverflow.in/502

If the binary tree in figure is traversed in inorder, then the order in which the nodes will be visited is ______



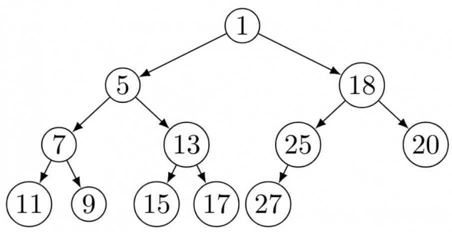
gate1991 binary-tree easy data-structures descriptive

Answer

3.6.10 Binary Tree: GATE CSE 1991 | Question: 14,a top

https://gateoverflow.in/541

Consider the binary tree in the figure below:



What structure is represented by the binary tree?

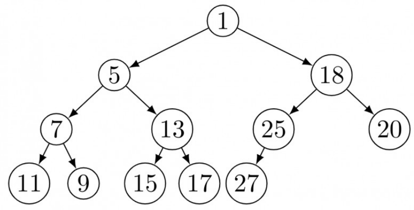
gate1991 data-structures binary-tree time-complexity normal descriptive

Answer

3.6.11 Binary Tree: GATE CSE 1991 | Question: 14,b top

https://gateoverflow.in/43026

Consider the binary tree in the figure below:



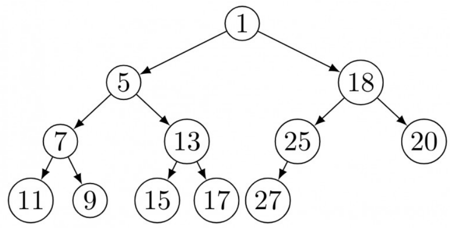
Give different steps for deleting the node with key 5 so that the structure is preserved.

gate1991 data-structures binary-tree normal descriptive

Answer

3.6.12 Binary Tree: GATE CSE 1991 | Question: 14,c top https://gateoverflow.in/43027

Consider the binary tree in the figure below:



Outline a procedure in Pseudo-code to delete an arbitrary node from such a binary tree with n nodes that preserves the structures. What is the worst-case time complexity of your procedure?

gate1991 normal data-structures binary-tree time-complexity descriptive

Answer

3.6.13 Binary Tree: GATE CSE 1993 | Question: 16 top https://gateoverflow.in/2313

Prove by the principal of mathematical induction that for any binary tree, in which every non-leaf node has 2-descendants, the number of leaves in the tree is one more than the number of non-leaf nodes.

gate1993 data-structures binary-tree normal descriptive

Answer

3.6.14 Binary Tree: GATE CSE 1994 | Question: 8 top https://gateoverflow.in/2504

A rooted tree with 12 nodes has its nodes numbered 1 to 12 in pre-order. When the tree is traversed in post-order, the nodes are visited in the order $3, 5, 4, 2, 7, 8, 6, 10, 11, 12, 9, 1$.

Reconstruct the original tree from this information, that is, find the parent of each node, and show the tree diagrammatically.

gate1994 data-structures binary-tree normal descriptive

Answer

3.6.15 Binary Tree: GATE CSE 1995 | Question: 1.17 top https://gateoverflow.in/2604

A binary tree T has n leaf nodes. The number of nodes of degree 2 in T is

A. $\log_2 n$
B. $n-1$
C. n
D. 2^n

gate1995 data-structures binary-tree normal

Answer

3.6.16 Binary Tree: GATE CSE 1995 | Question: 6 top https://gateoverflow.in/2667

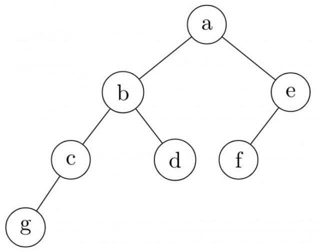
What is the number of binary trees with 3 nodes which when traversed in post-order give the sequence A, B, C? Draw all these binary trees.

gate1995 data-structures binary-tree normal descriptive

Answer

3.6.17 Binary Tree: GATE CSE 1996 | Question: 1.14 top https://gateoverflow.in/2718

In the balanced binary tree in the below figure, how many nodes will become unbalanced when a node is inserted as a child of the node "g"?



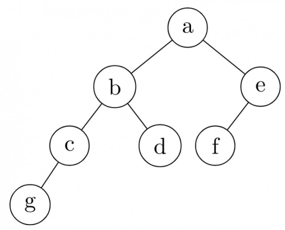
A. 1
B. 3
C. 7
D. 8

gate1996 data-structures binary-tree normal

Answer

3.6.18 Binary Tree: GATE CSE 1996 | Question: 1.15 top https://gateoverflow.in/2719

Which of the following sequences denotes the post order traversal sequence of the below tree?



A. $f\ e\ g\ c\ d\ b\ a$
B. $g\ c\ b\ d\ a\ f\ e$
C. $g\ c\ d\ b\ f\ e\ a$
D. $f\ e\ d\ g\ c\ b\ a$

gate1996 data-structures binary-tree easy

Answer

?

3.6.19 Binary Tree: GATE CSE 1997 | Question: 16 top https://gateoverflow.in/2276

A size-balanced binary tree is a binary tree in which for every node the difference between the number of nodes in the left and right subtree is at most 1. The distance of a node from the root is the length of the path from the root to the node. The height of a binary tree is the maximum distance of a leaf node from the root.

A. Prove, by using induction on h, that a size-balance binary tree of height h contains at least 2^h nodes.
B. In a size-balanced binary tree of height $h \geqslant 1$, how many nodes are at distance $h-1$ from the root? Write only the answer without any explanations.

Doubt in q.

gate1997 data-structures binary-tree normal descriptive proof

Answer

3.6.20 Binary Tree: GATE CSE 1998 | Question: 20 top https://gateoverflow.in/1734

Draw the binary tree with node labels a, b, c, d, e, f and g for which the inorder and postorder traversals result in the following sequences:

Inorder: a f b c d g e

Postorder: a f c g e d b

gate1998 data-structures binary-tree descriptive

Answer

3.6.21 Binary Tree: GATE CSE 2000 | Question: 1.14 top https://gateoverflow.in/637

Consider the following nested representation of binary trees: $(X\ Y\ Z)$ indicates Y and Z are the left and right subtrees, respectively, of node X. Note that Y and Z may be $NULL$, or further nested. Which of the following represents a valid binary tree?

A. (1 2 (4 5 6 7))
B. (1 (2 3 4) 5 6) 7)
C. (1 (2 3 4) (5 6 7))
D. (1 (2 3 $NULL$) (4 5))

gate2000-cse data-structures binary-tree easy

Answer

3.6.22 Binary Tree: GATE CSE 2000 | Question: 2.16 top https://gateoverflow.in/663

Let LASTPOST, LASTIN and LASTPRE denote the last vertex visited `in a postorder, inorder and preorder traversal respectively, of a complete binary tree. Which of the following is always true?

A. LASTIN = LASTPOST
B. LASTIN = LASTPRE
C. LASTPRE = LASTPOST
D. None of the above

gate2000-cse data-structures binary-tree normal

Answer

3.6.23 Binary Tree: GATE CSE 2002 | Question: 2.12 top https://gateoverflow.in/842

A weight-balanced tree is a binary tree in which for each node, the number of nodes in the left sub tree is at least half and at most twice the number of nodes in the right sub tree. The maximum possible height (number of nodes on the path from the root to the furthest leaf) of such a tree on n nodes is best described by which of the following?

A. $\log_2 n$
B. $\log_{\frac{4}{3}} n$
C. $\log_3 n$
D. $\log_{\frac{3}{2}} n$

gate2002-cse data-structures binary-tree normal

Answer

3.6.24 Binary Tree: GATE CSE 2002 | Question: 6 top https://gateoverflow.in/859

Draw all binary trees having exactly three nodes labeled A, B and C on which preorder traversal gives the sequence C, B, A.

gate2002-cse data-structures binary-tree easy descriptive

Answer

3.6.25 Binary Tree: GATE CSE 2004 | Question: 35 top https://gateoverflow.in/1032

Consider the label sequences obtained by the following pairs of traversals on a labeled binary tree. Which of these pairs identify a tree uniquely?

I. preorder and postorder
II. inorder and postorder
III. preorder and inorder
IV. level order and postorder

A. I only
B. II, III
C. III only
D. IV only

gate2004-cse data-structures binary-tree normal

Answer

3.6.26 Binary Tree: GATE CSE 2004 | Question: 43 top https://gateoverflow.in/1040

Consider the following C program segment

```
struct CellNode{
    struct CellNode *leftChild
    int element;
    struct CellNode *rightChild;
    };

int Dosomething (struct CellNode *ptr)
{
    int value = 0;
    if(ptr != NULL)
    {
        if (ptr -> leftChild != NULL)
            value = 1 + DoSomething (ptr -> leftChild);
        if (ptr -> rightChild != NULL)
            value = max(value, 1 + Dosomething (ptr -> rightChild));
    }
    return(value);
}
```

The value returned by the function DoSomething when a pointer to the root of a non-empty tree is passed as argument is

A. The number of leaf nodes in the tree
B. The number of nodes in the tree
C. The number of internal nodes in the tree
D. The height of the tree

gate2004-cse data-structures binary-tree normal

Answer

3.6.27 Binary Tree: GATE CSE 2006 | Question: 13 top https://gateoverflow.in/974

A scheme for storing binary trees in an array X is as follows. Indexing of X starts at 1 instead of 0. the root is stored at $X[1]$. For a node stored at $X[i]$, the left child, if any, is stored in $X[2i]$ and the right child, if any, in $X[2i+1]$. To be able to store any binary tree on n vertices the minimum size of X should be

A. $\log_2 n$
B. n
C. $2n+1$
D. 2^n-1

gate2006-cse data-structures binary-tree normal

Answer

3.6.28 Binary Tree: GATE CSE 2007 | Question: 12 top https://gateoverflow.in/1210

The height of a binary tree is the maximum number of edges in any root to leaf path. The maximum number of nodes in a binary tree of height h is:

A. $2^h - 1$
B. $2^{h-1} - 1$
C. $2^{h+1} - 1$
D. 2^{h+1}

gate2007-cse data-structures binary-tree easy

Answer

3.6.29 Binary Tree: GATE CSE 2007 | Question: 13 top https://gateoverflow.in/1211

The maximum number of binary trees that can be formed with three unlabeled nodes is:

A. 1
B. 5
C. 4
D. 3

gate2007-cse data-structures binary-tree normal

Answer

3.6.30 Binary Tree: GATE CSE 2007 | Question: 39, UGCNET-June2015-II: 22 top https://gateoverflow.in/1237

The inorder and preorder traversal of a binary tree are

d b e a f c g and a b d e c f g, respectively

The postorder traversal of the binary tree is:

A. d e b f g c a
B. e d b g f c a
C. e d b f g c a
D. d e f g b c a

gate2007-cse data-structures binary-tree normal ugcnetjune2015ii

Answer

3.6.31 Binary Tree: GATE CSE 2007 | Question: 46 top https://gateoverflow.in/1244

Consider the following C program segment where $CellNode$ represents a node in a binary tree:

```
struct CellNode {
    struct CellNode *leftChild;
    int element;
    struct CellNode *rightChild;
};

int Getvalue (struct CellNode *ptr) {
    int value = 0;
    if (ptr != NULL) {
        if ((ptr->leftChild == NULL) &&
            (ptr->rightChild == NULL))
            value = 1;
        else
            value = value + GetValue(ptr->leftChild)
                      + GetValue(ptr->rightChild);
    }
    return(value);
}
```

The value returned by $GetValue$ when a pointer to the root of a binary tree is passed as its argument is:

A. the number of nodes in the tree

B. the number of internal nodes in the tree
C. the number of leaf nodes in the tree
D. the height of the tree

gate2007-cse data-structures binary-tree normal

Answer

3.6.32 Binary Tree: GATE CSE 2010 | Question: 10 top

https://gateoverflow.in/2183

In a binary tree with n nodes, every node has an odd number of descendants. Every node is considered to be its own descendant. What is the number of nodes in the tree that have exactly one child?

A. 0
B. 1
C. $\frac{(n-1)}{2}$
D. $n-1$

gate2010-cse data-structures binary-tree normal

Answer

3.6.33 Binary Tree: GATE CSE 2011 | Question: 29 top

https://gateoverflow.in/2131

We are given a set of n distinct elements and an unlabeled binary tree with n nodes. In how many ways can we populate the tree with the given set so that it becomes a binary search tree?

A. 0
B. 1
C. $n!$
D. $\frac{1}{n+1} . {}^{2n}C_n$

gate2011-cse binary-tree normal

Answer

3.6.34 Binary Tree: GATE CSE 2012 | Question: 47 top

https://gateoverflow.in/2163

The height of a tree is defined as the number of edges on the longest path in the tree. The function shown in the pseudo-code below is invoked as height (root) to compute the height of a binary tree rooted at the tree pointer root.

```
int height(treeptr n)
{ if(n == NULL) return -1;
  if(n -> left == NULL)
     if(n -> right == NULL) return 0;
     else return B1; // Box 1

  else{h1 = height(n -> left);
       if(n -> right == NULL) return (1+h1);
       else{h2 = height(n -> right);
            return B2; // Box 2
            }
       }
}
```

The appropriate expressions for the two boxes **B1** and **B2** are:

A. **B1**: $(1 + \text{height}(n \rightarrow \text{ right}))$; **B2**: $(1 + \max(h1, h2))$
B. **B1**: $(\text{height}(n \rightarrow \text{ right}))$; **B2**: $(1 + \max(h1, h2))$
C. **B1**: $\text{height}(n \rightarrow \text{ right})$; **B2**: $\max(h1, h2)$
D. **B1**: $(1 + \text{height}(n \rightarrow \text{ right}))$; **B2**: $\max(h1, h2)$

gate2012-cse data-structures binary-tree normal

Answer

3.6.35 Binary Tree: GATE CSE 2014 Set 1 | Question: 12 top https://gateoverflow.in/1776

Consider a rooted n node binary tree represented using pointers. The best upper bound on the time required to determine the number of subtrees having exactly 4 nodes is $O(n^a \log^b n)$. Then the value of $a + 10b$ is __________.

gate2014-cse-set1 data-structures binary-tree numerical-answers normal

Answer

3.6.36 Binary Tree: GATE CSE 2015 Set 1 | Question: 25 top https://gateoverflow.in/8223

The height of a tree is the length of the longest root-to-leaf path in it. The maximum and minimum number of nodes in a binary tree of height 5 are

A. 63 and 6, respectively
B. 64 and 5, respectively
C. 32 and 6, respectively
D. 31 and 5, respectively

gate2015-cse-set1 data-structures binary-tree easy

Answer

3.6.37 Binary Tree: GATE CSE 2015 Set 2 | Question: 10 top https://gateoverflow.in/8059

A binary tree T has 20 leaves. The number of nodes in T having two children is ______.

gate2015-cse-set2 data-structures binary-tree normal numerical-answers

Answer

3.6.38 Binary Tree: GATE CSE 2015 Set 3 | Question: 25 top https://gateoverflow.in/8428

Consider a binary tree T that has 200 leaf nodes. Then the number of nodes in T that have exactly two children are ______.

gate2015-cse-set3 data-structures binary-tree normal numerical-answers

Answer

3.6.39 Binary Tree: GATE CSE 2016 Set 2 | Question: 36 top https://gateoverflow.in/39597

Consider the following New-order strategy for traversing a binary tree:

- Visit the root;
- Visit the right subtree using New-order;
- Visit the left subtree using New-order;

The New-order traversal of the expression tree corresponding to the reverse polish expression

```
3 4 * 5 - 2 ^ 6 7 * 1 + -
```

is given by:

A. $+ - 1\ 6\ 7 * 2 \wedge 5\ -\ 3\ 4 *$
B. $- + 1 * 6\ 7 \wedge 2 - 5 * 3\ 4$
C. $- + 1 * 7\ 6 \wedge 2\ -\ 5\ *\ 4\ 3$
D. $1\ 7\ 6 * + 2\ 5\ 4\ 3\ *\ \ - \wedge -$

gate2016-cse-set2 data-structures binary-tree normal

Answer

3.6.40 Binary Tree: GATE CSE 2018 | Question: 20 top https://gateoverflow.in/204094

The postorder traversal of a binary tree is $8, 9, 6, 7, 4, 5, 2, 3, 1$. The inorder traversal of the same tree is $8, 6, 9, 4, 7, 2, 5, 1, 3$. The height of a tree is the length of the longest path from the root to any leaf. The height of the binary tree above is _____ gate2018-cse data-structures binary-tree numerical-answers

Answer

3.6.41 Binary Tree: GATE CSE 2019 | Question: 46 top

☛ https://gateoverflow.in/302802

Let T be a full binary tree with 8 leaves. (A full binary tree has every level full.) Suppose two leaves a and b of T are chosen uniformly and independently at random. The expected value of the distance between a and b in T (ie., the number of edges in the unique path between a and b) is (rounded off to 2 decimal places) _________.

gate2019-cse numerical-answers data-structures binary-tree

Answer

3.6.42 Binary Tree: GATE CSE 2021 Set 2 | Question: 16 top

☛ https://gateoverflow.in/357524

Consider a complete binary tree with 7 nodes. Let A denote the set of first 3 elements obtained by performing Breadth-First Search (BFS) starting from the root. Let B denote the set of first 3 elements obtained by performing Depth-First Search (DFS) starting from the root.

The value of $|A-B|$ is _____________

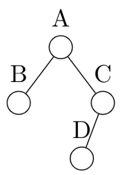
gate2021-cse-set2 numerical-answers data-structures binary-tree

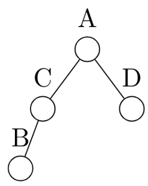
Answer

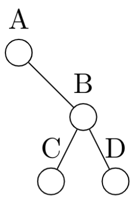
3.6.43 Binary Tree: GATE IT 2004 | Question: 54 top

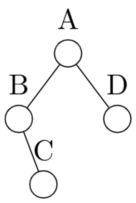
☛ https://gateoverflow.in/3697

Which one of the following binary trees has its inorder and preorder traversals as $BCAD$ and $ABCD$, respectively?

A. 

B. 

C. 

D. 

gate2004-it binary-tree easy data-structures

Answer

3.6.44 Binary Tree: GATE IT 2005 | Question: 50 top

☛ https://gateoverflow.in/3811

In a binary tree, for every node the difference between the number of nodes in the left and right subtrees is at most 2. If the height of the tree is $h > 0$, then the minimum number of nodes in the tree is

A. 2^{h-1}
B. $2^{h-1} + 1$
C. $2^h - 1$
D. 2^h

gate2005-it data-structures binary-tree normal

Answer

3.6.45 Binary Tree: GATE IT 2006 | Question: 71 top https://gateoverflow.in/3615

An array X of n distinct integers is interpreted as a complete binary tree. The index of the first element of the array is 0. The index of the parent of element $X[i], i \neq 0$, is?

A. $\left\lfloor \frac{i}{2} \right\rfloor$

B. $\left\lceil \frac{i-1}{2} \right\rceil$

C. $\left\lceil \frac{i}{2} \right\rceil$

D. $\left\lceil \frac{i}{2} \right\rceil - 1$

gate2006-it data-structures binary-tree normal

Answer

3.6.46 Binary Tree: GATE IT 2006 | Question: 73 top https://gateoverflow.in/3617

An array X of n distinct integers is interpreted as a complete binary tree. The index of the first element of the array is 0. If the root node is at level 0, the level of element $X[i]$, $i \neq 0$, is

A. $\lfloor \log_2 i \rfloor$
B. $\lceil \log_2 (i+1) \rceil$
C. $\lfloor \log_2 (i+1) \rfloor$
D. $\lceil \log_2 i \rceil$

gate2006-it data-structures binary-tree normal

Answer

3.6.47 Binary Tree: GATE IT 2006 | Question: 9 top https://gateoverflow.in/3548

In a binary tree, the number of internal nodes of degree 1 is 5, and the number of internal nodes of degree 2 is 10. The number of leaf nodes in the binary tree is

A. 10
B. 11
C. 12
D. 15

gate2006-it data-structures binary-tree normal

Answer

3.6.48 Binary Tree: GATE IT 2008 | Question: 46 top https://gateoverflow.in/3356

The following three are known to be the preorder, inorder and postorder sequences of a binary tree. But it is not known which is which.

I. $MBCAFHPYK$

II. $KAMCBYPFH$
III. $MABCKYFPH$

Pick the true statement from the following.

A. I and II are preorder and inorder sequences, respectively
B. I and III are preorder and postorder sequences, respectively
C. II is the inorder sequence, but nothing more can be said about the other two sequences
D. II and III are the preorder and inorder sequences, respectively

gate2008-it data-structures normal binary-tree

Answer

3.6.49 Binary Tree: GATE IT 2008 | Question: 76 top https://gateoverflow.in/3390

A binary tree with $n > 1$ nodes has n_1, n_2 and n_3 nodes of degree one, two and three respectively. The degree of a node is defined as the number of its neighbours.

n_3 can be expressed as

A. $n_1 + n_2 - 1$
B. $n_1 - 2$
C. $[((n_1 + n_2)/2)]$
D. $n_2 - 1$

gate2008-it data-structures binary-tree normal

Answer

3.6.50 Binary Tree: GATE IT 2008 | Question: 77 top https://gateoverflow.in/3391

A binary tree with $n > 1$ nodes has n_1, n_2 and n_3 nodes of degree one, two and three respectively. The degree of a node is defined as the number of its neighbours.

Starting with the above tree, while there remains a node v of degree two in the tree, add an edge between the two neighbours of v and then remove v from the tree. How many edges will remain at the end of the process?

A. $2 * n_1 - 3$
B. $n_2 + 2 * n_1 - 2$
C. $n_3 - n_2$
D. $n_2 + n_1 - 2$

gate2008-it data-structures binary-tree normal

Answer

Answers: Binary Tree

3.6.1 Binary Tree: GATE CSE 1987 | Question: 2c top https://gateoverflow.in/80579

✔ Yes it is possible since we can create Binary search tree , we know every Binary search tree is binary tree also but there are many binary tree possible , so we know that there is many binary tree possible without inorder.

So, answer is **NO** for this question

Refer: http://www.geeksforgeeks.org/if-you-are-given-two-traversal-sequences-can-you-construct-the-binary-tree/

References

👍 21 votes -- Prashant Singh (47.1k points)

3.6.2 Binary Tree: GATE CSE 1987 | Question: 2g top https://gateoverflow.in/80588

✔ Condition for binary tree is atmost two immediate child for every internal node

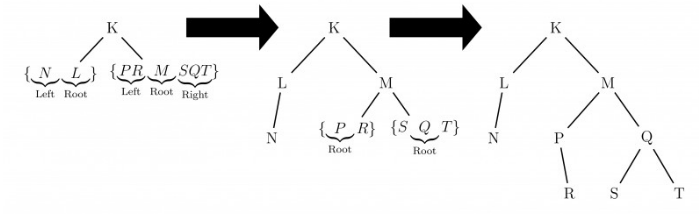
No, it is not the condition for binary tree.

21 votes -- Prashant Singh (47.1k points)

3.6.3 Binary Tree: GATE CSE 1987 | Question: 7b top https://gateoverflow.in/82427

✔ We can do as follows:

- Pre : $\overbrace{\text{K}}^{\text{Root}}$ L N M P R Q S T
- Post : $\underbrace{\text{N L}}_{\text{Left}}\ \underbrace{\text{K}}_{\text{Root}}\ \underbrace{\text{P R M S Q T}}_{\text{Right}}$



21 votes -- kirti singh (2.6k points)

3.6.4 Binary Tree: GATE CSE 1988 | Question: 7i top https://gateoverflow.in/94366

Height of a binary tree is the longest path from it's root to any of its leaves.

With number of nodes as n,

Max height possible: n-1 (skewed binary trees)

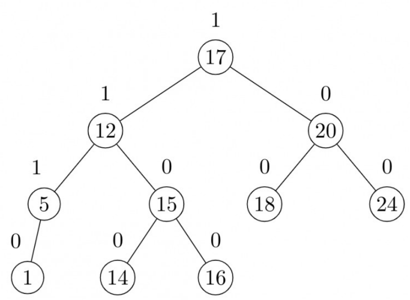
Min height possible: $floor(\log_2 n)$ (perfect binary trees)

An AVL tree is a height-balanced binary tree where the difference between the heights of the left subtree and the right subtree cannot exceed 1 for all nodes.

3 votes -- Ramyanee (131 points)

3.6.5 Binary Tree: GATE CSE 1988 | Question: 7ii top https://gateoverflow.in/94367

✔



Balancing factor $=$ the height of left subtree $-$ the height of right subtree
Balancing factors of all the nodes are marked in the figure.
Since there is no node that has a balancing factor greater than 1, we can say that the tree is balanced.

4 votes -- Akash (1.1k points)

Balance Factor $=$ height $($left Sub Tree$)-$ height $($right Sub Tree$)$

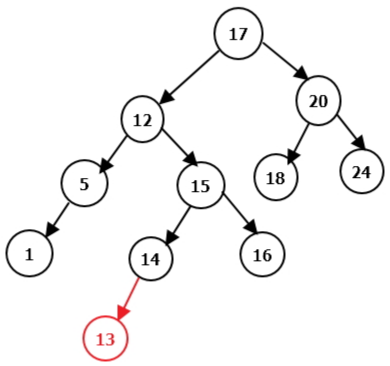
for height balance tree, balance factor of every node should be from $-1\ to\ 1\ \ i.e. (-1, 0, 1)$

So, given tree is height balanced tree.

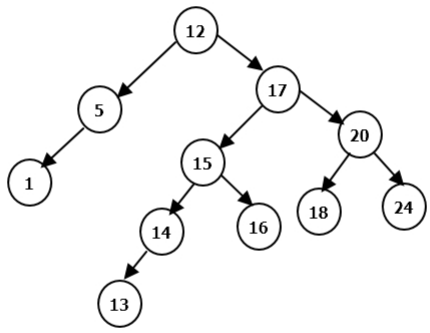
👍 3 votes -- Gurdeep (6.7k points)

3.6.6 Binary Tree: GATE CSE 1988 | Question: 7iii top ☛ https://gateoverflow.in/94368

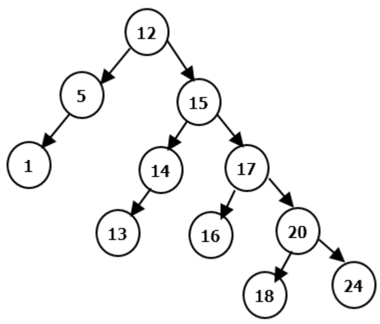
✔ 13 is less than 14, so it is placed on the left side of 14

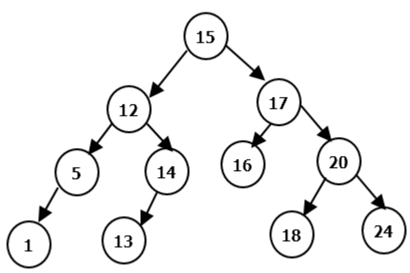


Now need to be rebalanced. Here 17 is unbalanced, So, in the 2^{nd} tree we need 1 right rotation to 17.



Now, 12 is unbalanced.To balance it we need 1 right rotation and 1 left rotation





👍 9 votes -- srestha (85.2k points)

3.6.7 Binary Tree: GATE CSE 1990 | Question: 3-iv top

https://gateoverflow.in/84828

✔ Here, n denotes the number of external (leaf) nodes and not the total number of nodes.

A. By adding an edge to the root of a skewed binary tree of say 10 nodes, we get a binary tree of 11 nodes having 2 external nodes of path lengths 9 and 1 respectively giving $\text{EPL} = 9 + 1 = 10 > 2^2$. So, option A is false.
B. This is always TRUE. The minimum EPL for a given number of external nodes n happens for a full binary tree. In this case when we have n external nodes each will have a path length of $\log_2 n$ giving $\text{EPL} = n \log_2 n$. Now, if we try to add any amount of skeweness to this full binary tree we can see that $\text{EPL} > n \log_2 n$.
C. False as shown for option A.
D. False as shown for option B.

Correct option: B.

2 votes -- Arjun Suresh (330k points)

3.6.8 Binary Tree: GATE CSE 1991 | Question: 01,viii top

https://gateoverflow.in/506

✔ This is straightforward. The nodes of the given tree are given in square boxes. The weights associated with the nodes are the numbers example $15, 9, 10$ etc.

Weighted path length $= \sum$ (for(each node in the tree) (path length) $*$(weight of the node)).

$$= \sum_{i=1}^{n} \text{Path Length}_i * \text{Weight of Node}_i$$

So answer (written in path_length * weight form) $= 4*2+4*4+4*5+4*7+3*9+3*10+1*15 = 144.$

52 votes -- arvchamp (131 points)

3.6.9 Binary Tree: GATE CSE 1991 | Question: 1,ix top

https://gateoverflow.in/502

✔ During the **in-order traversal** algorithm, the left subtree is explored first, followed by root, and finally nodes on the right subtree.

In order traversal is : 4 1 6 7 3 2 5 8.

22 votes -- Keith Kr (4.5k points)

3.6.10 Binary Tree: GATE CSE 1991 | Question: 14,a top

https://gateoverflow.in/541

✔ This is min-heap. It is obvious looking at the tree.
We cant use binary search as it is a heap.

29 votes -- Akash Kanase (36k points)

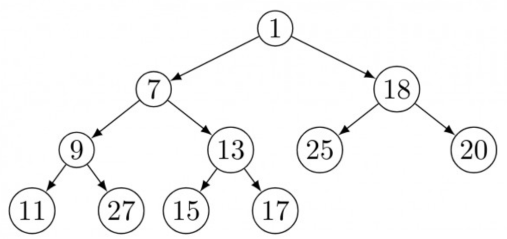
3.6.11 Binary Tree: GATE CSE 1991 | Question: 14,b top

https://gateoverflow.in/43026

✔ Since the given binary tree is a min-heap tree.

- First swap 27 and 5
- Then delete 5
- Apply min-heapify

And structure will be:



18 votes -- Manoj Kumar (26.7k points)

3.6.12 Binary Tree: GATE CSE 1991 | Question: 14,c top https://gateoverflow.in/43027

✔ By looking at the values it is clear that It is a **Min-Heap** Data structures. We know that Heap Data structures are stored in the array.

$\Longrightarrow$ **Delete procedure for Min-Heap Data Structure (If you already know the value and position of the node):**

1. Replace that node with the last element of that tree.
2. Apply Heapify property on that node.

For Example, Let If I want to delete 1, then I will replace that with 27. and apply heapify on that node. Or if i want to delete 5 then also I will replace that with 27, and apply heapify on that node.

Time Complexity: In this case, time complexity will not be more than $O(\log n)$.

$\Longrightarrow$ **Delete procedure for Min-Heap Data Structure (If you know the value but not position) :**

1. Find the position of the number by sequential search. (In the worst case it will take $O(n)$ time).
2. Replace that node with the last element of that tree.
3. Apply heapify property at that node.

Time Complexity: Wort time complexity of this algorithm will be $\mathbf{O(n + \log n)}$ **i.e.** $\mathbf{O(n)}$.

Note: This is a standard problem of Minimum element deletion from the Min-heap tree. The minimum element always resides at top (Root node). We just replace that value with the last element of the tree and apply heapify at the root node. The time complexity of that algorithm is $\mathbf{O(\log n)}$.

Here I have written the second method only to show that if we have to delete any of the nodes, and we just know the value but not the position. Since in question it is mentioned that **Arbitrary node**.

44 votes -- Muktinath Vishwakarma (23.9k points)

3.6.13 Binary Tree: GATE CSE 1993 | Question: 16 top https://gateoverflow.in/2313

✔ **Base Case :-** When we have just root then, there are no non leaf nodes. So No of leaves $= 1$, No of non leaf nodes is $= 0$. Base case holds.

Induction Hypothesis :- Assume that now for k internal nodes we will have $k + 1$ leaves.

Inducting on no of leaves, Now we add 2 more leaves to this tree. One of $k + 1$ leaf will become internal node. So now we will have $k + 1$ internal node. No of leafs will be $K + 1 - 1$ (1 leaf just became internal node) $+2$(New leafs) . So we proved that for any binary tree, in which every non-leaf node has 2-descendants, the number of leaves in the tree is one more than the number of non-leaf nodes.

16 votes -- Akash Kanase (36k points)

3.6.14 Binary Tree: GATE CSE 1994 | Question: 8 top https://gateoverflow.in/2504

✔

- PRE ORDER: $1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12$
- POST ORDER: $3, 5, 4, 2, 7, 8, 6, 10, 11, 12, 9, 1$

We can draw the tree, but in the question it is not specified if it is binary or ternary or something else.

Lets assume it is a binary tree.

Pre oder: Data, Left, Right (First node should be root)

(First node should be root, next to the root node should be left node of root, if in the post-order it is not in the second last position)

Post order: Left, Right, Data

(Last node should be root, before the last node it should be right node of root, if in pre-order it is not in the second position)

Now we can conclude that 9 is right of 1 and 2 is left of 1.

- PRE ORDER : $1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12$
- POST ORDER : $3, 5, 4, 2, 7, 8, 6, 10, 11, 12, 9$,$1$.

We can clearly observe that, the right of root contains $9, 10, 11, 12 \implies$ we can leave 9 as its position is fixed as immediate right of root.

So remaining elements are $10, 11, 12$.

What is pre-order of those elements ?

- $10, 11, 12$

What is the post order of those elements?

- $10, 11, 12$

Is it possible in Binary Tree? (check all 5 trees which can formed by 3 nodes)

NO

Now lets consider a Ternary Tree

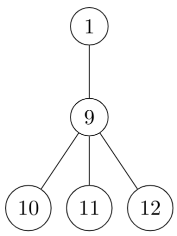
- PRE ORDER : $\overbrace{\boxed{1}}^{\text{root}}, \overbrace{2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12}^{\text{non-root elements}}$
- POST ORDER : $\underbrace{3, 5, 4, 2, 7, 8, 6, 10, 11, 12, 9}_{\text{non-root elements}}, \boxed{1}$

2 is left most and 9 is right most, the children of 9 are $10, 11, 12$ from left to right.

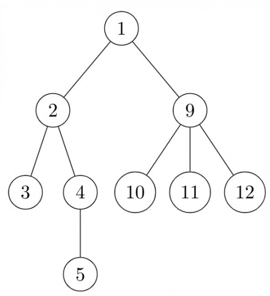
- PRE ORDER : $2, 3, 4, 5, 6, 7, 8, 9, \overbrace{\boxed{10, 11, 12}}^{\text{subtree of 9, but order unknown}}$
- POST ORDER : $3, 5, 4, 2, 7, 8, 6, 10, 11, 12, \overbrace{\boxed{9}}^{\text{Right most child of root}}$

Check all the elements in the preorder after $9-$ those should be children of 9.

After checking the preorder and postorder we can conclude that all those elements are at the same level in the order $10 - 11 - 12$.



- PRE ORDER : $\overbrace{\boxed{2}}^{\text{Should be the leftmost child of rootElements before 2 in the postordermust be children of 2}}, 3, 4, 5, 6, 7, 8$
- POST ORDER : $\underbrace{\boxed{3, 5, 4}}_{\text{Should be a subtree of 2but order in unknown}}, 2, 7, 8, 6$



How we separated $3, 4$ and 5 as two parts?

Check the preorder: $3 \implies$ elements before 3 in the postorder are in the same subtree as 3 but there are no elements before 3.

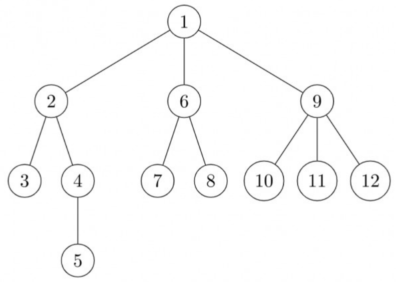
Therefore 3 is separated from 4 and 5.

How we fixed 4 and 5 in that order?

Check the preorder: $4 \implies$ elements before 4 in the postorder are in the same subtree of 4. Therefore 5 is in the same subtree of 4.

- PRE ORDER : $6, 7, 8$
- POST ORDER : $\underbrace{\boxed{7,8}}_{\text{Should be a sub-tree of 6but order is unknown}}$ $, 6$

We can easily understand that 6 is the child of root. Elements before 6 in the postorder forms subtree of 6.



How we fixed 7 and 8 in that order?

Check the preorder: $7 \implies$ elements before 7 in the postorder are in the same subtree of 7 but there are no elements before 7.

Therefore 7 and 8 are separated.

33 votes -- Shaik Masthan (50.4k points)

3.6.15 Binary Tree: GATE CSE 1995 | Question: 1.17 top https://gateoverflow.in/2604

✔ In Binary Tree a node can have at most 2 children.

Total number of node $\mathbf{N}$ = node with 0 child + node with 1 child + node with 2 child.

$\implies N = n_0 + n_1 + n_2$(here, in question it is given that no. of leaf nodes i.e no. of nodes with 0 children is n so $n_0 = n$)

$\implies N = n + n_1 + n_2$

Total number of edges $e = N - 1$, and also $e = n * 0 + n_1 * 1 + n_2 * 2$

$\therefore N - 1 = n * 0 + n_1 * 1 + n_2 * 2$

$\implies n + n_1 + n_2 - 1 = n_1 * 1 + n_2 * 2$

$\implies \mathbf{n_2 = n - 1}$

Option B is answer.

NOTE - For the tree, the d**egree** of a **node** is defined as the number of sub-trees of the **node or no of children of a node.**

71 votes -- Umang Raman (12.2k points)

3.6.16 Binary Tree: GATE CSE 1995 | Question: 6 top https://gateoverflow.in/2667

✔ There are only five such binary trees. This is given by 3^{rd} Catalan number as here we are finding the number of structurally similar binary trees with 3 nodes.

1. One with C as root and left child as A and right child B.
2. Second with C as root, B as left child and A as again left child of B.
3. Third with C as root, B as left child and A as right child of B.
4. Fourth with C as root, B as right child and A as right child of B.
5. Fifth with C as root, B as right child and A as left child of B.

References

29 votes -- Gate Keeda (15.9k points)

3.6.17 Binary Tree: GATE CSE 1996 | Question: 1.14 top https://gateoverflow.in/2718

(B).

a, b, c will become unbalanced with Balance factor as $+2, +2, +2$ respectively. Balance factor should be $-1, 0, +1$.

Balance factor = Height(LST) - Height(RST)

Or Balance factor = | Height(LST) - Height(RST) |

27 votes -- Gate Keeda (15.9k points)

3.6.18 Binary Tree: GATE CSE 1996 | Question: 1.15 top https://gateoverflow.in/2719

Correct Option: C

Left $\rightarrow$ Right $\rightarrow$ Root.

Ref: https://gateoverflow.in/2718/gate1996_1-14

References

18 votes -- Gate Keeda (15.9k points)

3.6.19 Binary Tree: GATE CSE 1997 | Question: 16 top https://gateoverflow.in/2276

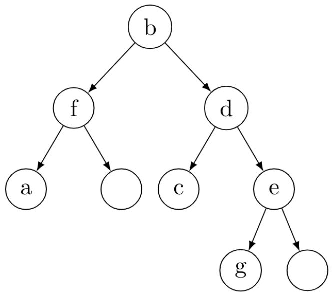
a. Prove, by using induction on h, that a size-balanced binary tree of height h contains at least 2^h nodes.
   When
   $h = 0$ .... least no. of nodes $= 2^0 = 1$
   $h = 1$ .... least no. of nodes $= 2^1 = 2$
   $h = 2$ .... least no. of nodes $= 2^2 = 4$
   Assume that the rule is true for $h = k$
   Then the min no. of nodes $= 2^k$ nodes
   If we increase the height by 1 by adding a node, we must also add nodes to fill the (max level -1) level.
   This would mean doubling the nodes
   Thus 2^{k+1}
   Hence, proved

b. In a size-balanced binary tree of height $h \geq 1$, how many nodes are at distance $h - 1$ from the root? Write only the answer
   without any explanation
   2^{h-1}

16 votes -- Sachin Mittal (15.8k points)

3.6.20 Binary Tree: GATE CSE 1998 | Question: 20 top https://gateoverflow.in/1734

The binary tree will be



👍 23 votes -- Anu (4.7k points)

3.6.21 Binary Tree: GATE CSE 2000 | Question: 1.14 top

☛ https://gateoverflow.in/637

✔
A. $\rightarrow$ $(4\ 5\ 6\ 7)$ this part of answer is not correct. We have $(X\ Y\ Z)$ not $(W\ X\ Y\ Z)$. So, this is wrong
B. $\rightarrow$ 3 closing paranthesis, 2 opening paranthesis. This is wrong.
C. CORRECT
D. $\rightarrow$ Here in $(1\ (2\ 3\ NULL)\ (4\ 5))$, $(4\ 5)$ this is not allowed. So this is wrong. (It should be $(4, 5, NULL)$)

👍 35 votes -- Akash Kanase (36k points)

3.6.22 Binary Tree: GATE CSE 2000 | Question: 2.16 top

☛ https://gateoverflow.in/663

✔ Inorder : Left $\rightarrow$ Root $\rightarrow$ Right

Preorder : Root $\rightarrow$ Left $\rightarrow$ Right

Postorder: Left $\rightarrow$ Right $\rightarrow$ Root

If the binary tree is full (last level is fully filled), the last visited node in Inorder and Preorder must be the rightmost one in the last level. But for a complete binary tree this need not be the case (in a complete binary tree last level need not be fully filled) and LASTPRE will be from the second last level in case the complete binary tree is not full. So, choice (**D**).

👍 53 votes -- Arjun Suresh (330k points)

3.6.23 Binary Tree: GATE CSE 2002 | Question: 2.12 top

☛ https://gateoverflow.in/842

✔ Let n_l and n_r nodes be present in the left and right sub-trees respectively.

We have, $\frac{n_r}{2} \leq n_l \leq 2n_r$. Without loss of generality, let the left sub-tree have greater number of nodes $(2n_r\ nodes)$. Then, $n_r + 2n_r + 1 = n$. Thus we get, $n_l = \frac{2(n-1)}{3}$ and $n_r = \frac{n-1}{3}$.

Total number of nodes can be described by the recurrence:

$T(n) = T\left(\frac{n-1}{3}\right) + T\left(\frac{2(n-1)}{3}\right) + 1$ and $T(1) = 1$

As this makes maximum nodes go to one subtree and that is what we want to get the maximum height with a given number of nodes.

Now, the height of the tree will be: $H(n) = H(\frac{2}{3}(n-1)) + 1$ and $H(1) = 1$

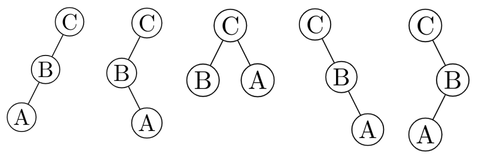
We can draw a recurrence tree and the cost at each level is 1, and the height will be $\log_{\frac{3}{2}}(n)$.
So, **D** option is the answer.

👍 84 votes -- Arjun Suresh (330k points)

3.6.24 Binary Tree: GATE CSE 2002 | Question: 6 top

☛ https://gateoverflow.in/859

✔



5 Binary trees.

👍 22 votes -- Anu (4.7k points)

3.6.25 Binary Tree: GATE CSE 2004 | Question: 35 top

☛ https://gateoverflow.in/1032

✔ **Following combination can uniquely identify a tree.**

- Inorder and Preorder.
- Inorder and Postorder.
- Inorder and Level-order.

And following do not.

- Postorder and Preorder.
- Preorder and Level-order.
- Postorder and Level-order.

Answer: **B**

👍 26 votes -- Shikhar Vashishth (3.1k points)

3.6.26 Binary Tree: GATE CSE 2004 | Question: 43 top

☛ https://gateoverflow.in/1040

✔ Correct Option: D

It calculates Height of tree.

Easy way to get this answer .

Draw a tree where all 4 parameters are different.

Get a Tree for which Height, No of Internal Nodes & No of Leafs are different & Trace out this algorithm.

👍 31 votes -- Akash Kanase (36k points)

3.6.27 Binary Tree: GATE CSE 2006 | Question: 13 top

☛ https://gateoverflow.in/974

✔ Answer is **D**.

> To be able to store **" any "** binary tree on n vertices the minimum size of X should be

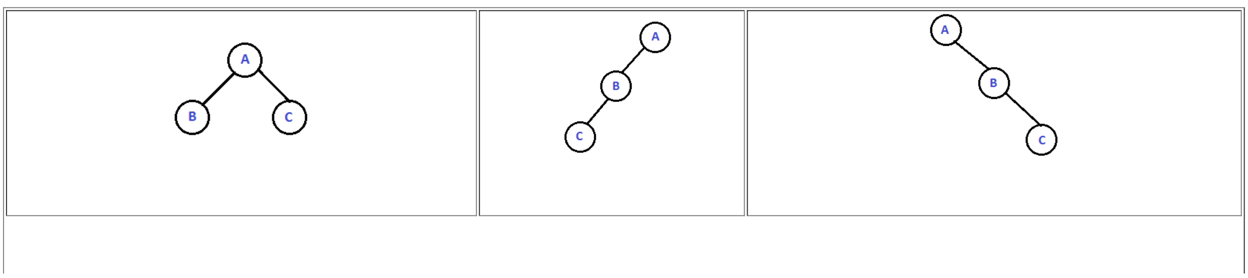
" Any Binary Tree and Size should be minimum " .
So We must consider worst case binary tree for this situation and find the minimum space required .

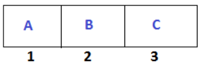
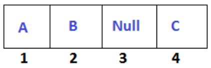
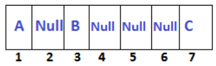
Minimum size for any binary tree

$\implies$ Minimum size of worst case binary tree

$X[i] = node X[2i] = \text{Left child} X[2i+1] = Rightchild$

Let $n = 3$



		
$X[1] = A$ $X[2] = B$ $X[3] = C$	$X[1] = A$ $X[2] = B$ $X[4] = C$	$X[1] = A$ $X[3] = B$ $X[7] = C$
n	2^{n-1}	$2^n - 1$
Minimum size Best Case binary tree		Minimum size Worst Case binary tree

73 votes -- Akhil Nadh PC (16.5k points)

Answer should be **(D)**.

Since binary tree can be of any form, the worst case happens for right skewed binary tree. Now, root goes to index 1, its child goes to index 3, its child goes to index 7 and so on the nth vertex goes to $2^n - 1$ th index of array.

34 votes -- Shaun Patel (6.1k points)

3.6.28 Binary Tree: GATE CSE 2007 | Question: 12 top https://gateoverflow.in/1210

✔ $2^{h+1} - 1$ just try this taking a small complete binary

never try to remember these formulae as remembering formulae is an overhead try to take examples in such cases.

Correct Answer: C

36 votes -- Bhagirathi Nayak (11.7k points)

3.6.29 Binary Tree: GATE CSE 2007 | Question: 13 top https://gateoverflow.in/1211

✔ Can be found with formula... $(2nCn/n+1)$... n being the number of nodes. for the given question... where $n = 3$... answer is 5. Let me also specify here.. that number of Binary Search Trees with n nodes is equal to number of unlabeled Binary trees.

http://gatecse.in/wiki/Number_of_Binary_trees_possible_with_n_nodes

Correct Answer: B

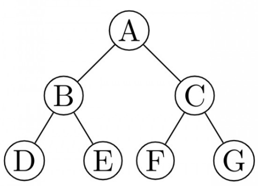
References

26 votes -- Gate Keeda (15.9k points)

3.6.30 Binary Tree: GATE CSE 2007 | Question: 39, UGCNET-June2015-II: 22 top https://gateoverflow.in/1237

✔ The answer is **A**.

Take the first node in preorder traversal - a will be the root of the tree
All nodes to the left of 'a' in inorder traversal will be in the left subtree of 'a' and all elements on the right will be in the right subtree of 'a'.
Take the second element from preorder traversal - 'b' - goes to left subtree of 'a' as it is in the left of 'a' in inorder list.
Proceeding likewise we can construct the binary tree as:



20 votes -- Gate Keeda (15.9k points)

3.6.31 Binary Tree: GATE CSE 2007 | Question: 46 top

https://gateoverflow.in/1244

✔ Answer: **C**

As the function returns 1 if and only if any node has both left & right children as $NULL$ (that node is a leaf node). Hence, value gets incremented at each leaf node.

👍 24 votes

-- Rajarshi Sarkar (27.8k points)

3.6.32 Binary Tree: GATE CSE 2010 | Question: 10 top

https://gateoverflow.in/2183

✔ 0 because every node has an odd number of descendants so least odd number 1 and every node is considered to be its own descendant so all nodes have even number of descendants $(0, 2, 4, 6 \dots)$ so every node has either 0 children or 2 children...

👍 45 votes

-- Murali (419 points)

3.6.33 Binary Tree: GATE CSE 2011 | Question: 29 top

https://gateoverflow.in/2131

✔ With n nodes, there are $\frac{^{2n}Cn}{(n+1)}$ distinct tree structures possible.

Corresponding to each structure, only one binary search tree (BST) can be formed because inorder is fixed.

Here, we are already given one such structure therefore only one tree possible.

If binary trees would have been asked, $n!$ trees would have been possible corresponding to each distinct tree structure. Here, tree structure is fixed and hence, we can have only one possibility for BST as elements are distinct. For general cases: http://gatecse.in/wiki/Number_of_Binary_trees_possible_with_n_nodes

Correct Answer: B

References

👍 67 votes

-- Anurag Semwal (6.7k points)

Given binary tree is unlabeled . So as it is given we are not allowed to change the formation of tree. Then To make it BST we can use atmost 1 way . As for particular structure we can not use n! arrangement of nodes (Becasue they are labeled and it is BST not BT)

👍 29 votes

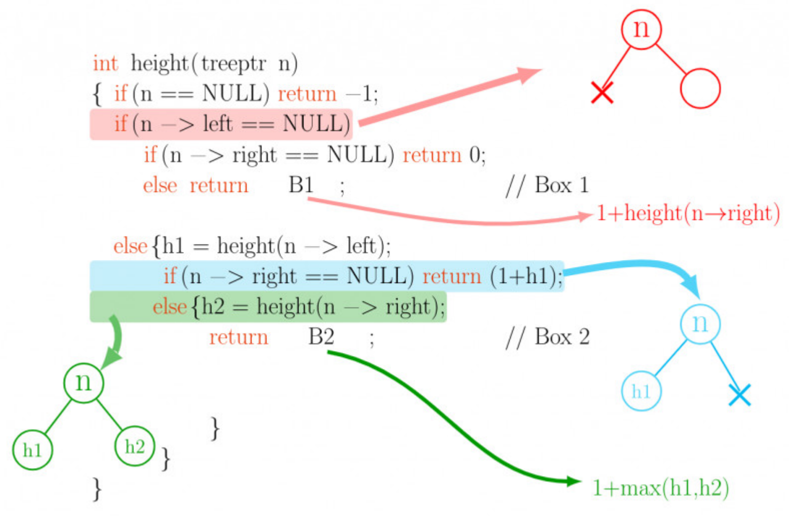
-- Palash (1.2k points)

3.6.34 Binary Tree: GATE CSE 2012 | Question: 47 top

https://gateoverflow.in/2163

✔ Answer is **option A.**

From the diagram below we are able to see how this works :



45 votes -- Amar Vashishth (25.2k points)

3.6.35 Binary Tree: GATE CSE 2014 Set 1 | Question: 12 top https://gateoverflow.in/1776

✔ Answer: 1..

Explanation:

1. Come to the 4th level up from the leaf node of the given binary tree, which can be done using tree traversal in $O(n)$.
2. For each node present in the level check whether it's subtree having exactly 4 nodes.. which can be done in constant time for each node, since it's subtree having constant number of nodes..
3. nodes in the level is less than n.. so its complexity is $O(n)$

Therefore, $a = 1$ and $b = 0$

$a + 10b = 1$... <- **Answer**

60 votes -- Vicky Bajoria (4.1k points)

We need to traverse all nodes at least once, and we need only one traversal. If $num(child1) + num(child2) + 1 = 4$, then output yes.

So, a must be 1 and $b = 0 \implies a + 10b = 1$.

75 votes -- Arjun Suresh (330k points)

3.6.36 Binary Tree: GATE CSE 2015 Set 1 | Question: 25 top https://gateoverflow.in/8223

✔ Option **A** is correct because height 5 means level 6 so maximum node $= 2^l - 1 = 2^6 - 1 = 63$ and for minimum, at each level only single node so total 6.

36 votes -- Anoop Sonkar (4.1k points)

3.6.37 Binary Tree: GATE CSE 2015 Set 2 | Question: 10 top https://gateoverflow.in/8059

✔

> In A binary tree if there are
> N leaf nodes then the number of nodes having 2 children will be N-1

Proof :

> Key idea is find number of edges using Degree and find number of edges using nodes and equate them

Let l be the number of leaf nodes, D_1 be the number of nodes with one child and D_2 be the number of nodes with two children.

Sum of Degrees, $D = 1 \times l + 3 \times D_2 + 2 \times D_1 - 1(\text{for root})$

(Root is having one degree less because it is not having a parent)

$D = l + 3D_2 + 2D_1 - 1 \qquad \rightarrow (1)$

Number of edges, $e = \frac{D}{2}$

Number of nodes, $n = D_2 + D_1 + l$

A tree with n nodes has $n - 1$ edges so,

$\frac{D}{2} = D_2 + D_1 + l - 1 \qquad \rightarrow (2)$

From (1) and (2)

$\frac{l+3D_2+2D_1-1}{2} = D_2 + D_1 + l - 1$

$\implies D_2 = l - 1$

So, the number of nodes in T having two children $= 20 - 1 = 19$

11 votes -- Rishi yadav (9k points)

3.6.38 Binary Tree: GATE CSE 2015 Set 3 | Question: 25 top

https://gateoverflow.in/8428

✔ Let number of nodes with exactly two children be x, and with exactly one children be y.

Total degree $= 200 + 3x + 2y - 1$ (As all nodes with 2 children have degree 3 except the root)

No. of nodes$= x + y + 200$

No. of edges $=$Total degree/2 $= (200 + 3x + 2y - 1)/2$ [Handshaking Theorem]

No. of edges in a tree $=$ No. of nodes-1

So, $(200 + 3x + 2y - 1) = 2x + 2y + 400 - 2$

$x = 199$

67 votes -- Arjun Suresh (330k points)

3.6.39 Binary Tree: GATE CSE 2016 Set 2 | Question: 36 top

https://gateoverflow.in/39597

✔ Expression given in reverse polish notation (i,e in Post-order)

convert first it into In-order

$3\;4\;*\;5\;-\;2\;\wedge\;6\;7\;*\;1\;+\;-$

$(3*4)\;5\;-\;2\;\wedge\;6\;7\;*\;1\;+\;-$

$((3*4)-5)\;2\;\wedge\;6\;7\;*\;1\;+\;-$

$(((3*4)-5)\wedge 2)\;6\;7\;*\;1\;+\;-$

$(((3*4)-5)\wedge 2)\;(6*7)\;1\;+\;-$

$(((3*4)-5)\wedge 2)\;((6*7)+1)\;-$

$((((3*4)-5)\wedge 2)-((6*7)+1))$

so Inorder expression in $((((3*4)-5)\wedge 2)-((6*7)+1))$

New-Order traversal is as by ROOT RIGHT LEFT

$((((3*4)-5)\wedge 2)-((6*7)+1))$

$-((6*7)+1)(((3*4)-5)\wedge 2)$

$-+1(6*7)(((3*4)-5)\wedge 2)$

$-+1*7\ 6(((3*4)-5)\wedge 2)$

$-+1*7\ 6\wedge 2((3*4)-5)$

$-+1*7\ 6\wedge 2-5(3*4)$

$-+1*7\ 6\wedge 2-5*4\ 3$

option C is correct

70 votes -- Praveen Saini (41.9k points)

It is quite simple actually.

Postorder: left, right, root

Neworder: Root, right, left

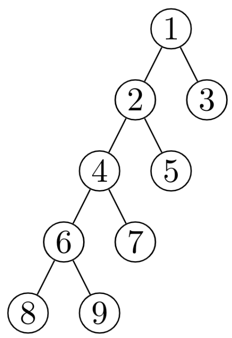
(left, right, root) and (Root, right, left) are reverse of each other, right ?

So, these two traversals will be exact reverse of each other! Option (c)!

159 votes -- Ashish Deshmukh (1.3k points)

3.6.40 Binary Tree: GATE CSE 2018 | Question: 20 top https://gateoverflow.in/204094

✔



Nodes in longest path from Root to Leaf $= (1-2, 2-4, 4-6, 6-8)$ *or* $(1-2, 2-4, 4-6, 6-9)$
$|\text{Longest Path}| = |(1-2, 2-4, 4-6, 6-8)| = 4$

26 votes -- Digvijay (44.9k points)

3.6.41 Binary Tree: GATE CSE 2019 | Question: 46 top https://gateoverflow.in/302802

✔

> Two leaves **a** and **b** of **T** are chosen uniformly and **independently** at random.

See the word "independently" here. It means that choice of a must not affect choice of b and vice versa. This implies that both a and b can even be the same node.

Now, we are given that the binary tree has 8 leaves. $(2^3 = 8) \implies$ we have 3 levels in the tree starting from 0. The distance in terms of number of edges between **any two leaf nodes will always be even**. If we consider any two leaves (not necessarily distinct)

- No. of pairs with path length $0 = 8$. (When $a = b$)
- No. of pairs with path length $2 = 8$. (For each of the 8 leaf nodes, we have a pair and since the selection is independent order of the pair is also significant)
- No. of pairs with path length $4 = 16$. (For each leaf node we have to go two levels up and while coming down we have 2

choices. So, we get $8 \times 2 = 16$ pairs)
- No. of pairs with path length $6 = 32$. (For each leaf node we have to go till the root, and from there while coming down it has $2 \times 2 = 4$ choices. Thus we get $8 \times 4 = 32$ pairs.

Total number of possible pairs $= 8 \times 8 = 64$

So, expected path length

$$= 0 \times \frac{8}{64} + 2 \times \frac{8}{64} + 4 \times \frac{16}{64} + 6 \times \frac{32}{64} = \frac{272}{64} = 4.25$$

80 votes -- Arjun Suresh (330k points)

3.6.42 Binary Tree: GATE CSE 2021 Set 2 | Question: 16 top https://gateoverflow.in/357524

✔ Complete binary tree property has this property that every level until last is fully filled and last level is filled from left to right.

So, when we have a complete binary tree with 7 nodes,

- level_1 has 1 node (root)
- level_2 has 2 nodes.
- level_3 has 4 nodes.

BFS goes level by level. So first three elements (say Set A) $= \text{level}_1$ nodes $(1) +$ $\text{level}_2(2)$ nodes.

DFS is go by connected manner. So first three elements (say Set B) $= \text{level}_1$ nodes (1)+ one node from level_2 nodes $+$ one node from level_3 nodes which is connected to the previously chosen level_2 node.
$A - B =$ the remaining node from the set of level_2 nodes.

$$\implies |A - B| = 1.$$

2 votes -- Shaik Masthan (50.4k points)

3.6.43 Binary Tree: GATE IT 2004 | Question: 54 top https://gateoverflow.in/3697

✔ Answer is **D.**
Inorder traversal is left node right.

Preorder is node left right.

24 votes -- Sankaranarayanan P.N (8.5k points)

3.6.44 Binary Tree: GATE IT 2005 | Question: 50 top https://gateoverflow.in/3811

Correct Option: B

Since the difference between the nodes in left and right subtree must hold for every node, until the last to last to last level, all levels must be fully filled. So, we get $2^{h-1} - 1$ nodes (No. of nodes in a complete binary tree of height $h - 2$). Now, our aim is to increase two more levels by adding minimum no. of nodes- just add two in nodes one below other to any of the nodes. So, we get $2^{h-1} + 1$ nodes.

46 votes -- Sneha Goel (819 points)

3.6.45 Binary Tree: GATE IT 2006 | Question: 71 top https://gateoverflow.in/3615

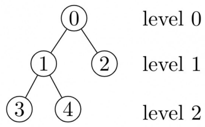
✔ Option is **(D).**

Left child of ith element will be at $2 * i + 1$ and right child at $2(i + 1)$

39 votes -- Sankaranarayanan P.N (8.5k points)

3.6.46 Binary Tree: GATE IT 2006 | Question: 73 top https://gateoverflow.in/3617

✔



PS : the value inside the node is array index

A. **FALSE** , $\lfloor \log_2 i \rfloor = \lfloor \log_2 1 \rfloor = 0$ but $X[1]$ is at level 1
B. **FALSE** , $\lceil \log_2 (i+1) \rceil = \lceil \log_2 (4+1) \rceil = \lceil \log_2 (5) \rceil = 3$ but $X[4]$ is at level 2
C. **Correct**
D. **FALSE** , $\lceil \log_2 i \rceil = \lceil \log_2 1 \rceil = 0$ but $X[1]$ is at level 1

21 votes -- Gate Ranker18 (2.4k points)

3.6.47 Binary Tree: GATE IT 2006 | Question: 9 top https://gateoverflow.in/3548

✔ A node in a binary tree has degree $0, 1$ or 2.

Ref: http://faculty.cs.niu.edu/~mcmahon/CS241/Notes/bintree.html

We are given no. of 1 degree node $= 5$, no. of 2 degree nodes $= 10$.

Total no. of edges $= 1 * 5 + 2 * 10 = 25$ (In tree degree is for outgoing edges only, and hence each degree corresponds to an edge)

So, total no. of nodes $= 25 + 1 = 26$ (No. of nodes in a tree is 1 more than no. of edges).

Now, no. of leaf nodes (nodes with 0 degree) $= 26 - 5 - 10 = 11$.

Correct Answer: B

References

84 votes -- Arjun Suresh (330k points)

3.6.48 Binary Tree: GATE IT 2008 | Question: 46 top https://gateoverflow.in/3356

✔ In preorder, root comes at the beginning of the traversal sequence and in postorder, root comes at the last of the traversal sequence. So, out of the given sequences only 1 and 2 are having such kind of order i.e K at the beginning and at the last.

Therefore, 2 is the preorder and 1 is postorder and the left sequence i.e 3 will definitely be inorder.

So, option **D** is correct.

36 votes -- Vivek sharma (2.1k points)

3.6.49 Binary Tree: GATE IT 2008 | Question: 76 top https://gateoverflow.in/3390

✔ Given definition of degree: no of neighbours of a node.

total nodes $= n = n_1 + n_2 + n_3$

Apply handshaking lemma:

Sum of degrees $= 2*$no of edges

$1 * n_1 + 2 * n_2 + 3 * n_3 = 2(n - 1)$

Total number of edges in graph will always be $(n_1 + n_2 + n_3 - 1)$.

$1 * n_1 + 2 * n_2 + 3 * n_3 = 2(n_1 + n_2 + n_3 - 1)$

$n_1 + 2n_2 + 3n_3 = 2n_1 + 2n_2 + 2n_3 - 2$

$n_3 = n_1 - 2$ **Option B**

66 votes -- swagnikd (561 points)

3.6.50 Binary Tree: GATE IT 2008 | Question: 77 top https://gateoverflow.in/3391

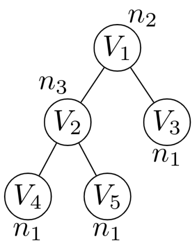
✔ Initially, $n_1 * 1 + n_2 * 2 + n_3 * 3 = 2(n_1 + n_2 + n_3 - 1) \implies n_3 = n_1 - 2$

Now we have removed all 2 degree nodes, so number of edges in final graph is $n_1 + n_3 - 1$.

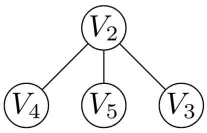
Put $n_3 = n_1 - 2$, we get

Number of edges $= 2 * n_1 - 3$

👍 74 votes -- Sumit1311 (1.4k points)



From the above tree, we will get the tree below



Now, check with the options we will get (A) as the answer.

👍 40 votes -- Shreyans Dhankhar (2.1k points)

3.7 Graph Search (1) top

3.7.1 Graph Search: GATE CSE 1989 | Question: 3-ixa top https://gateoverflow.in/87143

Which one of the following statements (s) is/are FALSE?

A. Overlaying is used to run a program, which is longer than the address space of the computer.
B. Optimal binary search tree construction can be performed efficiently by using dynamic programming.
C. Depth first search cannot be used to find connected components of a graph.
D. Given the prefix and postfix walls over a binary tree, the binary tree can be uniquely constructed.

normal gate1989 binary-tree graph-search multiple-selects

Answer

Answers: Graph Search

3.7.1 Graph Search: GATE CSE 1989 | Question: 3-ixa top https://gateoverflow.in/87143

A. **FALSE** according to definition of address space given in the link. whatever memory used by the overlay comes under the address space of computer "https://en.wikipedia.org/wiki/Address_space".
B. **TRUE** Optimal binary search tree construction can be performed efficiently by using dynamic programming. ref: http://www.geeksforgeeks.org/dynamic-programming-set-24-optimal-binary-search-tree/
C. **FALSE** Depth first search can be used to find connected components of a graph.
D. **FALSE** Infix + (postfix or prefix) is req. to construct the binary tree uniquely.

References

12 votes -- Lokesh Dafale (8.2k points)

3.8 Graphs (5) top

3.8.1 Graphs: GATE CSE 1997 | Question: 6.2 top https://gateoverflow.in/2258

Let G be the graph with 100 vertices numbered 1 to 100. Two vertices i and j are adjacent if $|i-j|=8$ or $|i-j|=12$. The number of connected components in G is

A. 8
B. 4
C. 12
D. 25

gate1997 data-structures normal graphs

Answer

3.8.2 Graphs: GATE CSE 2008 | Question: 42 top https://gateoverflow.in/1872

G is a graph on n vertices and $2n-2$ edges. The edges of G can be partitioned into two edge-disjoint spanning trees. Which of the following is NOT true for G?

A. For every subset of k vertices, the induced subgraph has at most $2k-2$ edges.
B. The minimum cut in G has at least 2 edges.
C. There are at least 2 edge-disjoint paths between every pair of vertices.
D. There are at least 2 vertex-disjoint paths between every pair of vertices.

gate2008-cse data-structures graphs normal

Answer

3.8.3 Graphs: GATE CSE 2014 Set 1 | Question: 3 top https://gateoverflow.in/1754

Let $G=(V,E)$ be a directed graph where V is the set of vertices and E the set of edges. Then which one of the following graphs has the same strongly connected components as G ?

A. $G_1=(V,E_1)$ where $E_1=\{(u,v)\mid(u,v)\notin E\}$
B. $G_2=(V,E_2)$ where $E_2=\{(u,v)\mid(v,u)\in E\}$
C. $G_3=(V,E_3)$ where $E_3=\{(u,v)|$ there is a path of length ≤ 2 from u to v in $E\}$
D. $G_4=(V_4,E)$ where V_4 is the set of vertices in G which are not isolated

gate2014-cse-set1 data-structures graphs ambiguous

Answer

3.8.4 Graphs: GATE CSE 2016 Set 1 | Question: 38 top https://gateoverflow.in/39731

Consider the weighted undirected graph with 4 vertices, where the weight of edge $\{i,j\}$ is given by the entry W_{ij} in the matrix W.

$$W=\begin{bmatrix}0&2&8&5\\2&0&5&8\\8&5&0&x\\5&8&x&0\end{bmatrix}$$

The largest possible integer value of x, for which at least one shortest path between some pair of vertices will contain the edge with weight x is ___________.

gate2016-cse-set1 data-structures graphs normal numerical-answers

Answer

3.8.5 Graphs: GATE IT 2008 | Question: 4 top https://gateoverflow.in/3264

What is the size of the smallest MIS (Maximal Independent Set) of a chain of nine nodes?

A. 5
B. 4
C. 3
D. 2

gate2008-it data-structures normal graphs

Answer

Answers: Graphs

3.8.1 Graphs: GATE CSE 1997 | Question: 6.2 top https://gateoverflow.in/2258

From the description it is clear that vertices are connected as follows:

$1-9-17-\ldots-97$
$2-10-18-\ldots-98$
$3-11-19-\ldots-99$
$4-12-20-\ldots-100$
$5-13-21-\ldots-93$
$6-14-22-\ldots-94$
$7-15-23-\ldots-95$
$8-16-24-\ldots-96$

We have covered all vertices using 8 vertex sets considering only $|i-j|=8$. Using $|i-j|=12$ we can see the vertex 1 is connected to $13, 2-14, 3-15$ and $4-16$, so the top 4 vertex sets are in fact connected to the bottom 4 sets, thus reducing the connected components to 4.

Correct Answer: B

62 votes -- Arjun Suresh (330k points)

3.8.2 Graphs: GATE CSE 2008 | Question: 42 top https://gateoverflow.in/1872

There are 2 spanning trees (a spanning tree connects all n vertices) for G which are edge disjoint. A spanning tree for n nodes require $n-1$ edges and so 2 edge-disjoint spanning trees requires $2n-2$ edges. As G has only $2n-2$ edges, it is clear that it doesn't have any edge outside that of the two spanning trees. Now lets see the cases:

Lets take any subgraph of G with k vertices. The remaining subgraph will have $n-k$ vertices. Between these two subgraphs (provided both has at least one vertex) there should be at least 2 edges, as we have 2 spanning trees in G. So, (B) is TRUE. Also, in the subgraph with k vertices, we cannot have more than $2k-2$ edges as this would mean that in the other subgraph with $n-k$ vertices, we would have less than $2n-2k$ edges, making 2 spanning trees impossible in it. So, (A) is also TRUE.

A spanning tree covers all the vertices. So, 2 edge-disjoint spanning trees in G means, between every pair of vertices in G we have two edge-disjoint paths (length of paths may vary). So, (C) is also TRUE.

So, that leaves option (**D**) as answer. It is not quite hard to give a counter example for (D).

49 votes -- Arjun Suresh (330k points)

3.8.3 Graphs: GATE CSE 2014 Set 1 | Question: 3 top https://gateoverflow.in/1754

(A) is false. Consider just two vertices connected to each other. So, we have one SCC. The new graph won't have any edges and so 2 SCC.

(**B**) is true. In a directed graph an SCC will have a path from each vertex to every other vertex. So, changing the direction of all the edges, won't change the SCC.

(D) is false. Consider any graph with isolated vertices- we loose those components.

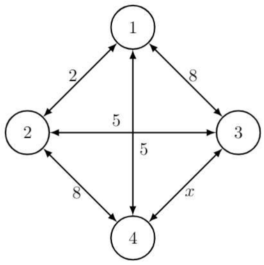
(C) is a bit tricky. Any edge is a path of length 1. So, the new graph will have all the edges from old one. Also, we are adding new edges (u, v). So, does this modify any SCC? No, because we add an edge (u, v), only if there is already a path of length $<= 2$ from u to v- so we do not create a new path. So, both (B) and (C) must answer, though GATE key says only B.

63 votes -- Arjun Suresh (330k points)

3.8.4 Graphs: GATE CSE 2016 Set 1 | Question: 38 top

https://gateoverflow.in/39731

✔



Let us list down the shortest edge between each pair of vertices x, y in the graph

x	y	shortest path from x to y
1	2	2
1	3	there is a direct path from 1 to 3 of weight 8.We can also choose to go via node 4 with total path weight $5 + x$. if $5 + x < 8 (x < 3)$ then shortest path is $5 + x$ otherwise shortest path is the weight of the direct path which is 8
1	4	5
2	3	5
2	4	there is a direct path from 2 to 4 of weight 8. We can also choose to go via node 3 with total path weight$5 + x$. If $5 + x < 8 (x < 3)$ the shortest path is 5+x. otherwise shortest path is the weight of the direct path which is 8
3	4	We can chose to go through the direct path of weight x or via nodes 2, 1 to node 4 with weight $5 + 2 + 5 = 12$. If $x < 12$ then we will chose x to be the shortest path otherwise 12 is the shortest path

- Case 1: Let us say $x < 3$. Say $x = 2$.

When we put $x = 2$ the above table is modified as

x	y	shortest path from x to y
1	2	2
1	3	7
1	4	5
2	3	5
2	4	7
3	4	2 // Note that the shortest path between nodes 3 and 4 is $x = 2$

- Case 2: $3 \leq x < 13$. Let's say $x = 12$. The table is now modified as

x	y	shortest path from x to y
1	2	2
1	3	8
1	4	5
2	3	5
2	4	8
3	4	12 // Note that the shortest path between nodes 3 and 4 is $x = 12$ and one of the shortest path is the direct edge x

Now the question asks you to find the largest possible integer value of x such that shortest path between at least one pair of nodes in the graph is equal to x. For values $x = 2, 3, 4, \ldots, 12$ the shortest path between node 3 and 4 is equal to x.

The largest among this is $x = 12$. So the answer is 12

PS: If the question is modified as "The largest possible integer value of x, for which the shortest path between **some pair of vertices is guaranteed** to contain the edge with weight x is"
then the answer will be 11.

Correct Answer: 12.

78 votes -- janakyMurthy (733 points)

3.8.5 Graphs: GATE IT 2008 | Question: 4 top https://gateoverflow.in/3264

✔ Answer: **C**

$1 - 2 - 3 - 4 - 5 - 6 - 7 - 8 - 9$

$(2, 5, 8)$ is the maximal independent set for a chain of 9 nodes. If we add any other node to the set then it will not be MIS.

39 votes -- Rajarshi Sarkar (27.8k points)

3.9 Hashing (16) top

3.9.1 Hashing: GATE CSE 1989 | Question: 1-vii, ISRO2015-14 top https://gateoverflow.in/10905

A hash table with ten buckets with one slot per bucket is shown in the following figure. The symbols $S1$ to $S7$ initially entered using a hashing function with linear probing. The maximum number of comparisons needed in searching an item that is not present is

0	S7
1	S1
2	
3	S4
4	S2
5	
6	S5
7	
8	S6
9	S3

A. 4
B. 5
C. 6
D. 3

hashing isro2015 gate1989 algorithms normal

Answer

3.9.2 Hashing: GATE CSE 1996 | Question: 1.13 top https://gateoverflow.in/2717

An advantage of chained hash table (external hashing) over the open addressing scheme is

A. Worst case complexity of search operations is less

B. Space used is less
C. Deletion is easier
D. None of the above

gate1996 data-structures hashing normal

Answer

3.9.3 Hashing: GATE CSE 1996 | Question: 15 top

https://gateoverflow.in/2767

Insert the characters of the string $K\ R\ P\ C\ S\ N\ Y\ T\ J\ M$ into a hash table of size 10.
Use the hash function

$$h(x) = (ord(x) – ord(\text{“}a\text{”}) + 1) \mod 10$$

and linear probing to resolve collisions.

A. Which insertions cause collisions?
B. Display the final hash table.

gate1996 data-structures hashing normal descriptive

Answer

3.9.4 Hashing: GATE CSE 1997 | Question: 12 top

https://gateoverflow.in/2272

Consider a hash table with n buckets, where external (overflow) chaining is used to resolve collisions. The hash function is such that the probability that a key value is hashed to a particular bucket is $\frac{1}{n}$. The hash table is initially empty and K distinct values are inserted in the table.

A. What is the probability that bucket number 1 is empty after the K^{th} insertion?
B. What is the probability that no collision has occurred in any of the K insertions?
C. What is the probability that the first collision occurs at the K^{th} insertion?

gate1997 data-structures hashing probability normal descriptive

Answer

3.9.5 Hashing: GATE CSE 2004 | Question: 7 top

https://gateoverflow.in/1004

Given the following input $(4322, 1334, 1471, 9679, 1989, 6171, 6173, 4199)$ and the hash function x mod 10, which of the following statements are true?

I. $9679, 1989, 4199$ hash to the same value
II. $1471, 6171$ hash to the same value
III. All elements hash to the same value
IV. Each element hashes to a different value

A. I only
B. II only
C. I and II only
D. III or IV

gate2004-cse data-structures hashing easy

Answer

3.9.6 Hashing: GATE CSE 2007 | Question: 40 top

https://gateoverflow.in/1238

Consider a hash table of size seven, with starting index zero, and a hash function $(3x+4) \mod 7$. Assuming the hash table is initially empty, which of the following is the contents of the table when the sequence $1, 3, 8, 10$ is inserted into the table using closed hashing? Note that $-$ denotes an empty location in the table.

A. 8, –, –, –, –, –, 10

B. $1, 8, 10, -, -, -, 3$
C. $1, -, -, -, -, -, 3$
D. $1, 10, 8, -, -, -, 3$

gate2007-cse data-structures hashing easy

Answer

3.9.7 Hashing: GATE CSE 2009 | Question: 36 top

☛ https://gateoverflow.in/1322

The keys $12, 18, 13, 2, 3, 23, 5$ and 15 are inserted into an initially empty hash table of length 10 using open addressing with hash function $h(k) = k \mod 10$ and linear probing. What is the resultant hash table?

A.

0	
1	
2	2
3	23
4	
5	15
6	
7	
8	18
9	

B.

0	
1	
2	12
3	13
4	
5	5
6	
7	
8	18
9	

C.

0	
1	
2	12
3	13
4	2
5	3
6	23
7	5
8	18
9	15

D.

0	
1	
2	2, 12
3	13, 3, 23
4	
5	5, 15
6	
7	
8	18
9	

gate2009-cse data-structures hashing normal

Answer

3.9.8 Hashing: GATE CSE 2010 | Question: 52 top https://gateoverflow.in/2360

A hash table of length 10 uses open addressing with hash function $h(k) = k \mod 10$, and linear probing. After inserting 6 values into an empty hash table, the table is shown as below

0	
1	
2	42
3	23
4	34
5	52
6	46
7	33
8	
9	

Which one of the following choices gives a possible order in which the key values could have been inserted in the table?

A. $46, 42, 34, 52, 23, 33$
B. $34, 42, 23, 52, 33, 46$
C. $46, 34, 42, 23, 52, 33$
D. $42, 46, 33, 23, 34, 52$

gate2010-cse data-structures hashing normal

Answer

3.9.9 Hashing: GATE CSE 2010 | Question: 53 top https://gateoverflow.in/43327

A hash table of length 10 uses open addressing with hash function $h(k) = k\,mod\,10$, and linear probing. After inserting 6 values into an empty hash table, the table is shown as below

0	
1	
2	42
3	23
4	34
5	52
6	46
7	33
8	
9	

How many different insertion sequences of the key values using the same hash function and linear probing will result in the hash table shown above?

A. 10
B. 20
C. 30
D. 40

data-structures hashing normal gate2010-cse

Answer

3.9.10 Hashing: GATE CSE 2014 Set 1 | Question: 40 top https://gateoverflow.in/1918

Consider a hash table with 9 slots. The hash function is $h(k) = k \mod 9$. The collisions are resolved by chaining.

The following 9 keys are inserted in the order: $5, 28, 19, 15, 20, 33, 12, 17, 10$. The maximum, minimum, and average chain lengths in the hash table, respectively, are

A. $3, 0$, and 1
B. $3, 3$, and 3
C. $4, 0$, and 1
D. $3, 0$, and 2

gate2014-cse-set1 data-structures hashing normal

Answer

3.9.11 Hashing: GATE CSE 2014 Set 3 | Question: 40 top

https://gateoverflow.in/2074

Consider a hash table with 100 slots. Collisions are resolved using chaining. Assuming simple uniform hashing, what is the probability that the first 3 slots are unfilled after the first 3 insertions?

A. $(97 \times 97 \times 97)/100^3$
B. $(99 \times 98 \times 97)/100^3$
C. $(97 \times 96 \times 95)/100^3$
D. $(97 \times 96 \times 95/(3! \times 100^3)$

gate2014-cse-set3 data-structures hashing probability normal

Answer

3.9.12 Hashing: GATE CSE 2015 Set 2 | Question: 33 top

https://gateoverflow.in/8152

Which one of the following hash functions on integers will distribute keys most uniformly over 10 buckets numbered 0 to 9 for i ranging from 0 to 2020?

A. $h(i) = i^2 \text{mod } 10$
B. $h(i) = i^3 \text{mod } 10$
C. $h(i) = (11 * i^2) \text{mod } 10$
D. $h(i) = (12 * i^2) \text{mod } 10$

gate2015-cse-set2 data-structures hashing normal

Answer

3.9.13 Hashing: GATE CSE 2015 Set 3 | Question: 17 top

https://gateoverflow.in/8414

Given that hash table T with 25 slots that stores 2000 elements, the load factor a for T is ________.

gate2015-cse-set3 data-structures hashing normal numerical-answers

Answer

3.9.14 Hashing: GATE IT 2006 | Question: 20 top

https://gateoverflow.in/3559

Which of the following statement(s) is TRUE?

I. A hash function takes a message of arbitrary length and generates a fixed length code.
II. A hash function takes a message of fixed length and generates a code of variable length.
III. A hash function may give the same hash value for distinct messages.

A. I only
B. II and III only
C. I and III only
D. II only

gate2006-it data-structures hashing normal

Answer

3.9.15 Hashing: GATE IT 2007 | Question: 28 top

https://gateoverflow.in/3461

Consider a hash function that distributes keys uniformly. The hash table size is 20. After hashing of how many keys will the probability that any new key hashed collides with an existing one exceed 0.5.

A. 5
B. 6
C. 7
D. 10

gate2007-it data-structures hashing probability normal

Answer

3.9.16 Hashing: GATE IT 2008 | Question: 48 top

https://gateoverflow.in/3358

Consider a hash table of size 11 that uses open addressing with linear probing. Let $h(k) = k \mod 11$ be the hash function used. A sequence of records with keys

43 36 92 87 11 4 71 13 14

is inserted into an initially empty hash table, the bins of which are indexed from zero to ten. What is the index of the bin into which the last record is inserted?

A. 3
B. 4
C. 6
D. 7

gate2008-it data-structures hashing normal

Answer

Answers: Hashing

3.9.1 Hashing: GATE CSE 1989 | Question: 1-vii, ISRO2015-14 top

https://gateoverflow.in/10905

✔ No of comparison in worst case for an element not in hash table is size of largest cluster $+1$. This is because the probe stops as soon as an empty slot is found (we r using linear probing here).

Size of largest cluster is 4 $(S6, S3, S7, S1)$

No of comparison is $4 + 1 = 5$

Correct Answer: B

76 votes

-- Digvijay (44.9k points)

3.9.2 Hashing: GATE CSE 1996 | Question: 1.13 top

https://gateoverflow.in/2717

✔

A. False :- search operation can go worst in chaining if all elements are stored under a single bucket.
B. False . Pointer space is overhead in chaining.
C. is true BCZ in Open Adressing sometimes though element is present we cant delete it if Empty Bucket comes in between while searching for that element ;Such Limitation is not there in Chaining.

26 votes

-- Rajesh Pradhan (18.9k points)

3.9.3 Hashing: GATE CSE 1996 | Question: 15 top

https://gateoverflow.in/2767

✔ Here $Order(x) - Order('a')$ means count the number of characters between character $'x'$ and $'a'$.

Assuming $a = 0, b = 1$ & so on.

a. J & M cause collusion.
b. Final Hash Table

Index	key
0	T
1	K
2	J
3	C
4	N
5	Y
6	P
7	M
8	R
9	S

20 votes -- Akash Kanase (36k points)

3.9.4 Hashing: GATE CSE 1997 | Question: 12 top https://gateoverflow.in/2272

✔

A. Probability that buckets other than 1 are selected $= \dfrac{n-1}{n}$

This should happen k times and each of the k events are independent so $\dfrac{(n-1)^k}{n^k}$

B. When $k=1$, probability of no collision $= \dfrac{n}{n}$ (for only one insert, there can be no collission)

For $k=2$ probability of no collision $= \dfrac{n}{n} \times \dfrac{n-1}{n}$

For $k=n$ probability of no collision $= \dfrac{n}{n} \times \dfrac{n-1}{n} \times \dfrac{n-2}{n} \times \ldots \times \dfrac{n-k+1}{n}$ for $k \leq n$

For $k > n$ probability of no collision $= 0$ (even though we are using chaining without a collision a chain cannot start)

C. Probability of first collision at $(k=1) = \dfrac{k-1}{n}$

Probability of first collision at $(k=2) = \dfrac{n}{n} \times \dfrac{k-1}{n}$

Probability of first collision at $(k=3) = \dfrac{n}{n} \times \dfrac{n-1}{n} \times \dfrac{k-1}{n}$

Probability of first collision at $(k \leq n) = \dfrac{n}{n} \times \dfrac{n-1}{n} \times \dfrac{n-2}{n} \times \ldots \times \dfrac{n-k+2}{n} \times \dfrac{k-1}{n}$

Probability of first collision at $(k=n+1) = \dfrac{n}{n} \times \dfrac{n-1}{n} \times \dfrac{n-2}{n} \times \ldots \times \dfrac{1}{n} \times \dfrac{n}{n}$

For $k > n+1$ probability of first collision $= 0$ (as it should have definitely happened in one of the previous $(n+1)$ insertions.

38 votes -- Danish (3.4k points)

3.9.5 Hashing: GATE CSE 2004 | Question: 7 top https://gateoverflow.in/1004

✔ Option **C** is correct answer because the last digit of every digit given is equal in I and II.

16 votes -- Bhagirathi Nayak (11.7k points)

3.9.6 Hashing: GATE CSE 2007 | Question: 40 top https://gateoverflow.in/1238

✔ The answer is (**B**).

1 will occupy location $0, 3$ will occupy location $6, 8$ hashed to location 0 which is already occupied so, it will be hashed to one location next to it. i.e. to location 1.

Since 10 also clashes, so it will be hashed to location 2.

22 votes -- Gate Keeda (15.9k points)

3.9.7 Hashing: GATE CSE 2009 | Question: 36 top https://gateoverflow.in/1322

✔ (**C**) is the correct option ..directly from the definition of linear probing. In linear probing, when a hashed location is already filled, locations are linearly probed until a free one is found.

http://courses.cs.washington.edu/courses/cse326/00wi/handouts/lecture16/sld015.htm

References

29 votes -- Bhagirathi Nayak (11.7k points)

3.9.8 Hashing: GATE CSE 2010 | Question: 52 top https://gateoverflow.in/2360

✔ Option (**C**)

$46, 34, 42, 23, 52, 33$

- $46-$ position 6
- 34 position 4
- 42 position 2
- 23 position 3
- 52 position $2-$ collision next empty is 5
- 33 position $3-$ collision next empty is 7

19 votes -- Sankaranarayanan P.N (8.5k points)

3.9.9 Hashing: GATE CSE 2010 | Question: 53 top https://gateoverflow.in/43327

✔ 53 - option (**C**).

Slots $3, 4, 5$ and 6 must be filled before 33 comes. Similarly slots $2, 3$ and 4 must be filled before 52 comes. And 52 must come before 33, as it is not occupying slot 2. So, 33 must be at the end and 52 can come at position 4 or 5.

Let 52 come at position 4. Slots $2, 3$ and 4 must be filled before 52 leaving only slot 6 left for the element coming at position 5 which should be 46. So, the first 3 elements can come in any order giving $3! = 6$ ways.

Let 52 come at position 5. Now, the first four elements can come in any order. So, $4! = 24$ ways.

So, total number of different insertion sequences possible $= 24 + 6 = 30$

67 votes -- Arjun Suresh (330k points)

answer = option C

the element 33 has managed to hold position at slot #7 it means elements should occupy slot #3 to slot #6 before it in the sequence. Currently, it seems like all element except 42 should come before 33 in the sequence.

But, element 52 requires that 42 comes before it. and 33 requires that 52 comes before it. This means that 42 has to come before 33. this makes element 33 to occupy last position in the sequence.

now for element 52 to occupy its place these can be two cases :

- Case 1 : $\boxed{\{42, 23, 34\}}$, 52
- Case 2: $\boxed{\{42, 23, 34, 46\}}$, 52

Case 1 means that those three elements comes before $52 = 3! = 6$ways
Case 2 means that those four elements comes before $52 = 4! = 24$ways

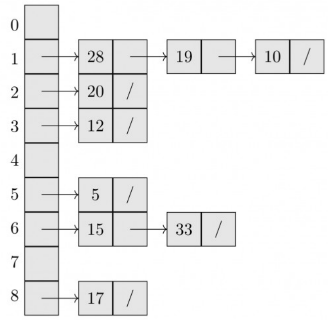
Combining all info we get,
Total number of sequences possible that will form the same hash table as above $= (6 + 24) \times 1 = 30$

👍 61 votes -- Amar Vashishth (25.2k points)

3.9.10 Hashing: GATE CSE 2014 Set 1 | Question: 40 top

https://gateoverflow.in/1918

✔



So, Maximum & minimum chain lengths are 3 & 0 respectively.

Average chain length $= (0+3+1+1+0+1+2+0+1)/9 = 1$.

So, Answer is A.

👍 40 votes -- Jay (831 points)

3.9.11 Hashing: GATE CSE 2014 Set 3 | Question: 40 top

https://gateoverflow.in/2074

✔ We have 100 slots each of which are picked with equal probability by the hash function (since hashing is uniform). So, to avoid first 3 slots, the hash function has to pick from the remaining 97 slots. And repetition is allowed, since chaining is used- meaning a list of elements are stored in a slot and not a single element.

So, required probability $= \frac{97}{100} \times \frac{97}{100} \times \frac{97}{100}$

$= (97 \times 97 \times 97)/100^3$

Correct Answer: A

👍 88 votes -- Arjun Suresh (330k points)

3.9.12 Hashing: GATE CSE 2015 Set 2 | Question: 33 top

https://gateoverflow.in/8152

✔ Since mod 10 is used, the last digit matters.

If we CUBE all numbers from 0 to 9, we get the following

Number	Cube	Last Digit in Cube
0	0	0
1	1	1
2	8	8
3	27	7
4	64	4
5	125	5
6	216	6
7	343	3
8	512	2
9	729	9

Therefore all numbers from 0 to 2020 are equally divided in to 10 buckets. If we make a table for square, we won't get equal distribution as shown in the following table. $1, 4, 6$ and 9 are repeated, so these buckets would have more entries and there are no buckets corresponding to $2, 3, 7$ and 8.

Number	Square	Last Digit in Cube
0	0	0
1	1	**1**
2	4	**4**
3	9	**9**
4	16	**6**
5	25	5
6	36	**6**
7	49	**9**
8	64	**4**
9	81	**1**

http://geeksquiz.com/gate-gate-cs-2015-set-2-question-43/

Correct Answer: B

References

87 votes -- Anu (4.7k points)

3.9.13 Hashing: GATE CSE 2015 Set 3 | Question: 17 top https://gateoverflow.in/8414

✔ A critical statistic for a hash table is the **load factor**, that is the number of entries divided by the number of buckets:

Load factor $= n/k$

where:

$n =$ number of entries

$k =$ number of buckets

As the load factor grows larger, the hash table becomes slower, and it may even fail to work (depending on the method used).

Here, load factor = $2000/25 = 80$

33 votes -- Akash Kanase (36k points)

3.9.14 Hashing: GATE IT 2006 | Question: 20 top https://gateoverflow.in/3559

✔ Answer is (**C**).

I. A hash function takes a message of arbitrary length and generates a fixed length code.. This is correct, this is directly from definition of hash function. Ref: https://en.wikipedia.org/wiki/Hash_function

II. As I is correct II is wrong !

III. This is true. example: Hash function $N\%10$, this will generate same values for 1 as well as 11!

(Even in cryptographic hash functions collision happens, just that it is not easy to find colliding instances!)

References

30 votes -- Akash Kanase (36k points)

3.9.15 Hashing: GATE IT 2007 | Question: 28 top https://gateoverflow.in/3461

✔ The question is a bit ambiguous.

> After hashing of how many keys, will the probability that any new key hashed collides with an existing one exceed 0.5.

Here, 'new key hashed' is the ambiguity. It can mean the probability of a collision in the next 'hash', or the probability of a collision in any of the hashes of the 'new keys' starting from the first insertion. For the first case answer must be 10 to get probability equal to 0.5, and so 11 must be the answer for probability > 0.5. Thus we can conclude from given choices, it is the second case.

So, we need to find n such that after n hashes, probability of collision (in any of the n hashes) > 0.5.

Probability that there will be a collision after n hashes (a collision happened in at least one of those n hashes)
$= 1-$Probability that there was no collision in the first n hashes

$= 1 - 1.\frac{19}{20}.\frac{18}{20} \dots \frac{20-n+1}{20}.$

So, we need,

$0.5 < 1 - 1.\frac{19}{20}.\frac{18}{20} \dots \frac{20-n+1}{20}.$

$\implies \frac{19}{20}.\frac{18}{20} \dots \frac{20-n+1}{20} < 0.5.$

For $n = 5$, we get, 0.5814 and for $n = 6$, we get 0.43605. So, answer should be $n = 6$.

Correct Answer: B

69 votes -- Arjun Suresh (330k points)

3.9.16 Hashing: GATE IT 2008 | Question: 48 top https://gateoverflow.in/3358

✔

Index	key
0	87
1	11
2	13
3	36
4	92
5	4
6	71
7	**14**
8	
9	
10	43

(D) is answer

17 votes -- Prashant Singh (47.1k points)

3.10 Heap (24) top

3.10.1 Heap: GATE CSE 1990 | Question: 2-viii top https://gateoverflow.in/83993

Match the pairs in the following questions:

(a)	A heap construction	(p)	$\Omega(n \log_{10} n)$
(b)	Constructing Hashtable with linear probing	(q)	$O(n)$
(c)	AVL tree construction	(r)	$O(n^2)$
(d)	Digital trie construction	(s)	$O(n \log_{10} n)$

gate1990 match-the-following data-structures heap

Answer

3.10.2 Heap: GATE CSE 1996 | Question: 2.11 top https://gateoverflow.in/2740

The minimum number of interchanges needed to convert the array into a max-heap is

$89, 19, 40, 17, 12, 10, 2, 5, 7, 11, 6, 9, 70$

A. 0
B. 1
C. 2
D. 3

gate1996 data-structures heap easy

Answer

3.10.3 Heap: GATE CSE 1999 | Question: 12 top https://gateoverflow.in/1511

A. In binary tree, a full node is defined to be a node with 2 children. Use induction on the height of the binary tree to prove that the number of full nodes plus one is equal to the number of leaves.
B. Draw the min-heap that results from insertion of the following elements in order into an initially empty min-heap: $7, 6, 5, 4, 3, 2, 1$. Show the result after the deletion of the root of this heap.

gate1999 data-structures heap normal descriptive

Answer

3.10.4 Heap: GATE CSE 2001 | Question: 1.15 top https://gateoverflow.in/708

Consider any array representation of an n element binary heap where the elements are stored from index 1 to index n of the array. For the element stored at index i of the array $(i \leq n)$, the index of the parent is

A. $i - 1$
B. $\lfloor \frac{i}{2} \rfloor$
C. $\lceil \frac{i}{2} \rceil$
D. $\frac{(i+1)}{2}$

gate2001-cse data-structures heap easy

Answer

3.10.5 Heap: GATE CSE 2003 | Question: 23 top https://gateoverflow.in/1110

In a min-heap with n elements with the smallest element at the root, the 7^{th} smallest element can be found in time

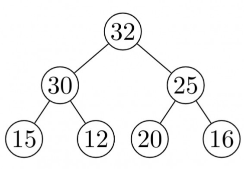
A. $\Theta(n \log n)$
B. $\Theta(n)$
C. $\Theta(\log n)$
D. $\Theta(1)$

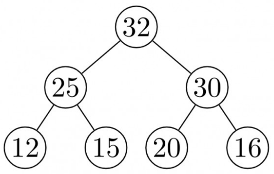
gate2003-cse data-structures heap

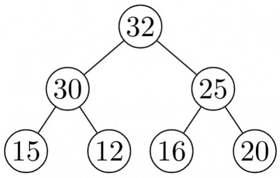
Answer

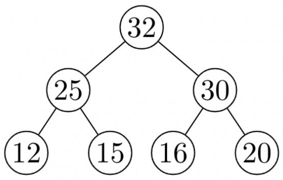
3.10.6 Heap: GATE CSE 2004 | Question: 37 top https://gateoverflow.in/1034

The elements $32, 15, 20, 30, 12, 25, 16,$ are inserted one by one in the given order into a maxHeap. The resultant maxHeap is

A. 

B. 

C. 

D. 

gate2004-cse data-structures heap normal

Answer

3.10.7 Heap: GATE CSE 2005 | Question: 34 top

https://gateoverflow.in/1370

A priority queue is implemented as a Max-Heap. Initially, it has 5 elements. The level-order traversal of the heap is: $10, 8, 5, 3, 2$. Two new elements 1 and 7 are inserted into the heap in that order. The level-order traversal of the heap after the insertion of the elements is:

A. $10, 8, 7, 5, 3, 2, 1$
B. $10, 8, 7, 2, 3, 1, 5$
C. $10, 8, 7, 1, 2, 3, 5$
D. $10, 8, 7, 3, 2, 1, 5$

gate2005-cse data-structures heap normal

Answer

3.10.8 Heap: GATE CSE 2006 | Question: 10 top

https://gateoverflow.in/889

In a binary max heap containing n numbers, the smallest element can be found in time

A. $O(n)$
B. $O(\log n)$
C. $O(\log \log n)$
D. $O(1)$

gate2006-cse data-structures heap easy

Answer

3.10.9 Heap: GATE CSE 2006 | Question: 76 top

https://gateoverflow.in/1852

Statement for Linked Answer Questions 76 & 77:

A 3-ary max heap is like a binary max heap, but instead of 2 children, nodes have 3 children. A 3-ary heap can be represented by an array as follows: The root is stored in the first location, $a[0]$, nodes in the next level, from left to right, is stored from $a[1]$ to $a[3]$. The nodes from the second level of the tree from left to right are stored from $a[4]$ location onward. An item x can be inserted into a 3-ary heap containing n items by placing x in the location $a[n]$ and pushing it up the tree to satisfy the heap property.

76. Which one of the following is a valid sequence of elements in an array representing 3-ary max heap?

A. $1, 3, 5, 6, 8, 9$

B. $9, 6, 3, 1, 8, 5$
C. $9, 3, 6, 8, 5, 1$
D. $9, 5, 6, 8, 3, 1$

gate2006-cse data-structures heap normal

Answer

3.10.10 Heap: GATE CSE 2006 | Question: 77 top

https://gateoverflow.in/87191

Statement for Linked Answer Questions 76 & 77:

A 3-ary max heap is like a binary max heap, but instead of 2 children, nodes have 3 children. A 3-ary heap can be represented by an array as follows: The root is stored in the first location, $a[0]$, nodes in the next level, from left to right, is stored from $a[1]$ to $a[3]$. The nodes from the second level of the tree from left to right are stored from $a[4]$ location onward. An item x can be inserted into a 3-ary heap containing n items by placing x in the location $a[n]$ and pushing it up the tree to satisfy the heap property.

76. Which one of the following is a valid sequence of elements in an array representing $3-$ary max heap?

A. $1, 3, 5, 6, 8, 9$
B. $9, 6, 3, 1, 8, 5$
C. $9, 3, 6, 8, 5, 1$
D. $9, 5, 6, 8, 3, 1$

77. Suppose the elements $7, 2, 10$ and 4 are inserted, in that order, into the valid 3-ary max heap found in the previous question, Q.76. Which one of the following is the sequence of items in the array representing the resultant heap?

A. $10, 7, 9, 8, 3, 1, 5, 2, 6, 4$
B. $10, 9, 8, 7, 6, 5, 4, 3, 2, 1$
C. $10, 9, 4, 5, 7, 6, 8, 2, 1, 3$
D. $10, 8, 6, 9, 7, 2, 3, 4, 1, 5$

gate2006-cse data-structures heap normal

Answer

3.10.11 Heap: GATE CSE 2007 | Question: 47 top

https://gateoverflow.in/1245

Consider the process of inserting an element into a $Max\ Heap$, where the $Max\ Heap$ is represented by an $array$. Suppose we perform a binary search on the path from the new leaf to the root to find the position for the newly inserted element, the number of $comparisons$ performed is:

A. $\Theta(\log_2 n)$

B. $\Theta(\log_2 \log_2 n)$

C. $\Theta(n)$

D. $\Theta(n \log_2 n)$

gate2007-cse data-structures heap normal

Answer

3.10.12 Heap: GATE CSE 2009 | Question: 59 top

https://gateoverflow.in/1341

Consider a binary max-heap implemented using an array.
Which one of the following array represents a binary max-heap?

A. $\{25, 12, 16, 13, 10, 8, 14\}$
B. $\{25, 14, 13, 16, 10, 8, 12\}$
C. $\{25, 14, 16, 13, 10, 8, 12\}$
D. $\{25, 14, 12, 13, 10, 8, 16\}$

gate2009-cse data-structures heap normal

Answer

3.10.13 Heap: GATE CSE 2009 | Question: 60 top

https://gateoverflow.in/43466

Consider a binary max-heap implemented using an array.
What is the content of the array after two delete operations on $\{25, 14, 16, 13, 10, 8, 12\}$

A. $\{14, 13, 12, 10, 8\}$
B. $\{14, 12, 13, 8, 10\}$
C. $\{14, 13, 8, 12, 10\}$
D. $\{14, 13, 12, 8, 10\}$

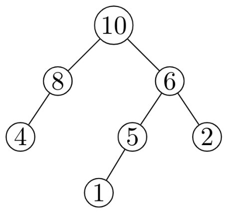
gate2009-cse data-structures heap normal

Answer

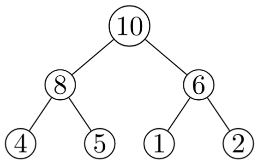
3.10.14 Heap: GATE CSE 2011 | Question: 23 top

https://gateoverflow.in/2125

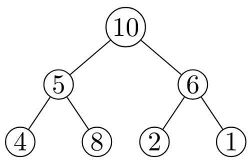
A max-heap is a heap where the value of each parent is greater than or equal to the value of its children. Which of the following is a max-heap?



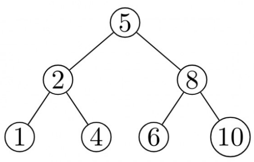
A.



B.



C.



D.

gate2011-cse data-structures heap easy

Answer

3.10.15 Heap: GATE CSE 2014 Set 2 | Question: 12 top

https://gateoverflow.in/1967

A priority queue is implemented as a Max-Heap. Initially, it has 5 elements. The level-order traversal of the heap is: $10, 8, 5, 3, 2$. Two new elements 1 and 7 are inserted into the heap in that order. The level-order traversal of the heap after the insertion of the elements is:

A. $10, 8, 7, 3, 2, 1, 5$
B. $10, 8, 7, 2, 3, 1, 5$
C. $10, 8, 7, 1, 2, 3, 5$
D. $10, 8, 7, 5, 3, 2, 1$

gate2014-cse-set2 data-structures heap normal

Answer

3.10.16 Heap: GATE CSE 2015 Set 1 | Question: 32 top

☛ https://gateoverflow.in/8273

Consider a max heap, represented by the array: $40, 30, 20, 10, 15, 16, 17, 8, 4$.

Array index	1	2	3	4	5	6	7	8	9
Value	40	30	20	10	15	16	17	8	4

Now consider that a value 35 is inserted into this heap. After insertion, the new heap is

A. $40, 30, 20, 10, 15, 16, 17, 8, 4, 35$
B. $40, 35, 20, 10, 30, 16, 17, 8, 4, 15$
C. $40, 30, 20, 10, 35, 16, 17, 8, 4, 15$
D. $40, 35, 20, 10, 15, 16, 17, 8, 4, 30$

gate2015-cse-set1 data-structures heap easy

Answer

3.10.17 Heap: GATE CSE 2015 Set 2 | Question: 17 top

☛ https://gateoverflow.in/8091

Consider a complete binary tree where the left and right subtrees of the root are max-heaps. The lower bound for the number of operations to convert the tree to a heap is

A. $\Omega(\log n)$
B. $\Omega(n)$
C. $\Omega(n \log n)$
D. $\Omega(n^2)$

gate2015-cse-set2 data-structures heap normal

Answer

3.10.18 Heap: GATE CSE 2015 Set 3 | Question: 19 top

☛ https://gateoverflow.in/8418

Consider the following array of elements.

$\langle 89, 19, 50, 17, 12, 15, 2, 5, 7, 11, 6, 9, 100 \rangle$

The minimum number of interchanges needed to convert it into a max-heap is

A. 4
B. 5
C. 2
D. 3

gate2015-cse-set3 data-structures heap normal

Answer

3.10.19 Heap: GATE CSE 2016 Set 1 | Question: 37 top

☛ https://gateoverflow.in/39706

An operator $delete(i)$ for a binary heap data structure is to be designed to delete the item in the i-th node. Assume that the heap is implemented in an array and i refers to the i-th index of the array. If the heap tree has depth d (number of edges on the path from the root to the farthest leaf), then what is the time complexity to re-fix the heap efficiently after the removal of the element?

A. $O(1)$
B. $O(d)$ but not $O(1)$
C. $O(2^d)$ but not $O(d)$
D. $O(d\,2^d)$ but not $O(2^d)$

gate2016-cse-set1 data-structures heap normal

Answer

3.10.20 Heap: GATE CSE 2016 Set 2 | Question: 34 top

https://gateoverflow.in/39585

A complete binary min-heap is made by including each integer in $[1, 1023]$ exactly once. The depth of a node in the heap is the length of the path from the root of the heap to that node. Thus, the root is at depth 0. The maximum depth at which integer 9 can appear is _________.

gate2016-cse-set2 data-structures heap normal numerical-answers

Answer

3.10.21 Heap: GATE CSE 2019 | Question: 40 top

https://gateoverflow.in/302808

Consider the following statements:

I. The smallest element in a max-heap is always at a leaf node
II. The second largest element in a max-heap is always a child of a root node
III. A max-heap can be constructed from a binary search tree in $\Theta(n)$ time
IV. A binary search tree can be constructed from a max-heap in $\Theta(n)$ time

Which of te above statements are TRUE?

A. I, II and III
B. I, II and IV
C. I, III and IV
D. II, III and IV

gate2019-cse data-structures heap

Answer

3.10.22 Heap: GATE IT 2004 | Question: 53 top

https://gateoverflow.in/3696

An array of integers of size n can be converted into a heap by adjusting the heaps rooted at each internal node of the complete binary tree starting at the node $\lfloor (n-1)/2 \rfloor$, and doing this adjustment up to the root node (root node is at index 0) in the order $\lfloor (n-1)/2 \rfloor$, $\lfloor (n-3)/2 \rfloor$, ..., 0. The time required to construct a heap in this manner is

A. $O(\log n)$
B. $O(n)$
C. $O(n \log \log n)$
D. $O(n \log n)$

gate2004-it data-structures heap normal

Answer

3.10.23 Heap: GATE IT 2006 | Question: 44 top

https://gateoverflow.in/3587

Which of the following sequences of array elements forms a heap?

A. $\{23, 17, 14, 6, 13, 10, 1, 12, 7, 5\}$
B. $\{23, 17, 14, 6, 13, 10, 1, 5, 7, 12\}$
C. $\{23, 17, 14, 7, 13, 10, 1, 5, 6, 12\}$
D. $\{23, 17, 14, 7, 13, 10, 1, 12, 5, 7\}$

gate2006-it data-structures heap easy

Answer

3.10.24 Heap: GATE IT 2006 | Question: 72 top

https://gateoverflow.in/3616

An array X of n distinct integers is interpreted as a complete binary tree. The index of the first element of the array is 0. If only the root node does not satisfy the heap property, the algorithm to convert the complete binary tree into a heap has the best asymptotic time complexity of

A. $O(n)$
B. $O(\log n)$
C. $O(n \log n)$
D. $O(n \log \log n)$

gate2006-it data-structures heap easy

Answer

Answers: Heap

3.10.1 Heap: GATE CSE 1990 | Question: 2-viii top

https://gateoverflow.in/83993

✔ (A) A heap Construction(Build heap) : -

Build-Heap (A)

1. heap-size$[A] \leftarrow$ length$[A]$
2. for $i \leftarrow \lfloor \text{length}[A]/2 \rfloor$ downto 1
3. do Heapify (A, i)

Note that $A[(\lfloor n/2 \rfloor + 1) \dots n]$ are all leaves so they are length 1 heaps.

Trivial Analysis: Each call to Heapify requires $\log(n)$ time, we make n such calls $\implies O(n \log n)$.

Tighter Bound: Each call to Max-Heapify requires time $O(h)$ where h is the height of node i. Therefore running time is

$$\sum_{h=0}^{\log n} \frac{h}{2^h+1} \times O(h) = O\left(n \sum_{h=0}^{\log n} \frac{h}{2^h}\right)$$

where $2^h + 1 =$ Number of nodes at height h and $O(h) =$ Running time for each node.

$$= O\left(n \sum_{h=0}^{\infty} \frac{h}{2^h}\right) = O(n)$$

Note: $\sum_{h=0}^{\infty} \frac{h}{2^h} = 2$

Reference:

- https://courses.csail.mit.edu/6.006/fall10/handouts/recitation10-8.pdf
- http://www.cs.fsu.edu/~burmeste/slideshow/chapter7/7-3.html

(B) Constructing Hash table with linear probing for n elements has time complexity $\implies O(n^2)$ in the worst case. This happens when all the n elements are hashed to the same location and this means the number of probes will go like $0, 1, 2, 3, \dots n-1$ for the n elements giving the total number of probes $\frac{n.(n-1)}{2}$ which is $O(n^2)$.

(C) AVL Tree construction : -

- AVL trees are height-balanced binary search trees.
- Balance factor of a node : height(left subtree) $-$ height(right subtree).
- An AVL tree has balance factor calculated at every node, for every node, heights of left and right subtree can differ by no more than 1.
- Constructing an AVL tree, to create a tree we have to insert n elements in it. To insert the element in a balanced tree we need $\log(n)$. Therefore we can do this with $O(n \log(n))$ time complexity.
- We can traverse an AVL tree in $O(n)$ time and the inorder traversal gives the sorted list of n elements. Since, sorting of n elements takes minimum $\Omega(n \log n)$ time this also implies that creation of AVL tree is $\Omega(n \log n)$.
- From above two points we get the time complexity of AVL tree creation as $\Theta(n \log n)$

Reference:

- https://stackoverflow.com/questions/17629668/difference-between-the-time-complexity-required-to-build-binary-search-tree-and
- https://courses.cs.washington.edu/courses/cse373/06sp/handouts/lecture12.pdf
- https://www.cs.auckland.ac.nz/software/AlgAnim/AVL.html

(D) Digital trie construction of n keys with maximum length m requires $O(n \times m)$ time. Among the options m is not specified and assuming m is constant (small length keys), the time complexity will be $O(n)$ which is also $O(n \log n)$. So, (s) gets mapped to (D) and (p) to (C) though both (p) and (s) are possible for (C).

Correct Answer:

$A-q, B-r, C-p, D-s$

References

👍 11 votes -- Lakshman Patel (64.9k points)

3.10.2 Heap: GATE CSE 1996 | Question: 2.11 top

https://gateoverflow.in/2740

✔ "The minimum number of interchanges needed to convert the array
$89, 19, 40, 17, 12, 10, 2, 5, 7, 11, 6, 9, 70$
into a heap with the maximum element at the root node is:"

This is the correction.

Answer: **C**.

Only element 70 violates the rule. Hence, it must be shifted to its proper position.
Step1: $swap(10, 70)$
Step2: $swap(40, 70)$
Hence, only 2 interchanges are required.

👍 24 votes -- Gate Keeda (15.9k points)

3.10.3 Heap: GATE CSE 1999 | Question: 12 top

https://gateoverflow.in/1511

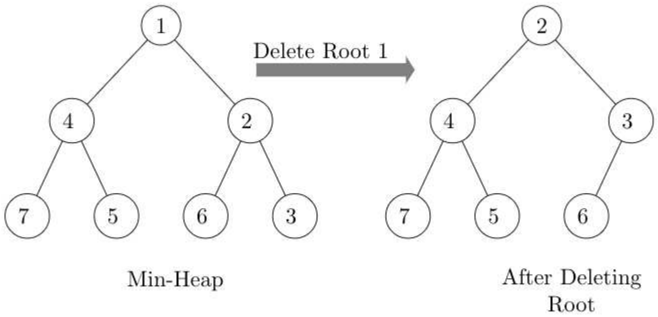
✔

a. Note My Convention:-
   Number of full nodes $= F$
   Number of leaf node $= L$

   Base Case: $H = 0$.
   A binary tree of height 0 is just a single node with no children, and therefore has 1 leaf.
   $F + 1 = L$
   $0 + 1 = 1$
   so the base case satisfies the induction hypothesis (see below).
   Induction Hypothesis (I.H.): Suppose that for some $k \geq 0$, all binary trees of height $\leq k$ have $(F + 1) = L$ leaves .
   Induction Step: Let T be a binary tree of height $k + 1$. Then $T's$ left and right subtrees are each binary trees of height $\leq k$, and thus by the I.H. both subtree have $(F + 1)$ leaves. The number of leaves in T is equal to the sum of the number of leaves in T's subtrees,
   $(F + 1)$ (left sub tree) $+(F + 1)$ (right sub tree) $= L$ (left sub tree) $+L$ (right sub tree)
   $2F + 2 = 2L$
   $2(F + 1) = 2(L)$
   $\therefore F + 1 = L$ (proved)

   Hence, the hypothesis holds for $k + 1$, and so the theorem is proved.

b. Here in question they mentioned to insert element in given order into an empty Heap.
   So here we have to use Insertion Method to create the heap instead of using Heapify Method to build the heap.
   Please refer the below image where the LHS shows the resultant heap after doing insertion of the keys into initial empty heap.
   RHS heap is the result of deletion of root.



👍 24 votes -- Rajesh Pradhan (18.9k points)

3.10.4 Heap: GATE CSE 2001 | Question: 1.15 top

☛ https://gateoverflow.in/708

✔ for node at index i

left $child(L)$ at 2i

right $child(R)$ at $2i+1$

for node at index i

parent will be at floor $i/2$

Correct Answer: B

👍 27 votes -- Pooja Palod (24.1k points)

3.10.5 Heap: GATE CSE 2003 | Question: 23 top

☛ https://gateoverflow.in/1110

✔ Time to find the smallest element on a min-heap- one retrieve operation - $\Theta(1)$
Time to find the second smallest element on a min-heap- requires $2^2-1=3$ check operations to find the second smallest element out of 3 elements - $\Theta(1)$

Time to find the 7^{th} smallest element - requires $O(2^7-1)=O(127)$ check operations to find the seventh smallest element out of 127 possible ones - $\Theta(1)$

In short if the number of required operations is independent of the input size n, then it is always $\Theta(1)$.

(Here, we are doing a level order traversal of the heap and checking the elements)

If we are not allowed to traverse the heap and allowed only default heap-operations, we will be restricted with doing Extract-min 7 times which would be $O(\log n)$.

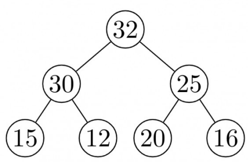
Correct Answer: D

👍 98 votes -- gatecse (62.6k points)

3.10.6 Heap: GATE CSE 2004 | Question: 37 top

☛ https://gateoverflow.in/1034

✔ The answer is **option A.**



Just keep inserting elements making sure resulting Tree is nearly Complete. (Heap Property) .
While inserting any node, if you find that Value of New Node $>$ Value of its parent, bubble it up to keep Max heap property

👍 22 votes -- Akash Kanase (36k points)

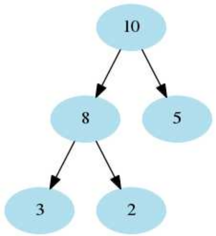
3.10.7 Heap: GATE CSE 2005 | Question: 34 top

☛ https://gateoverflow.in/1370

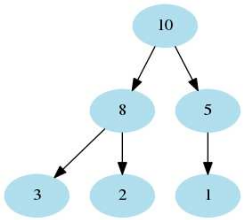
✔ Answer is (**D**)

Whenever we insert an element in heap, it will be in last level from left to right. So, here we insert element 1 and 7 as children of node 5. Then we perform heapify algorithm until we get the min/max heap. So, finally we get the heap whose level order traversal is $10, 8, 7, 3, 2, 1, 5$

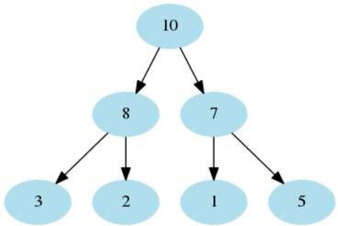
Initial heap:



After insert of 1



After insert of 7



👍 19 votes -- neha pawar (3.3k points)

3.10.8 Heap: GATE CSE 2006 | Question: 10 top

☛ https://gateoverflow.in/889

✔

A. $O(n)$

In a max heap, the smallest element is always present at a leaf node. Heap being a complete binary tree, there can be up to $\frac{n}{2}$ leaf nodes and to examine all of them we would need $O(n)$ time.

👍 40 votes -- Keith Kr (4.5k points)

3.10.9 Heap: GATE CSE 2006 | Question: 76 top

☛ https://gateoverflow.in/1852

✔ D.

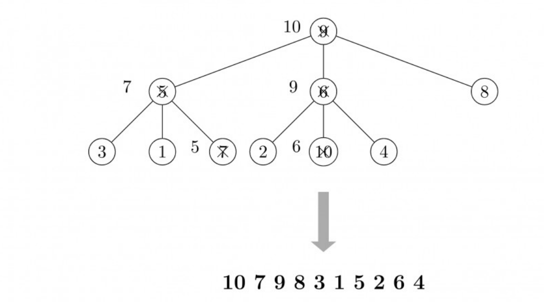
The method to solve is clearly given in the question itself. Just one thing to mention is the heap property. Max Heap = The parent is always greater or equal to the child. if not then swap the respective child with the parent to satisfy the property of the heap.

👍 11 votes -- Gate Keeda (15.9k points)

3.10.10 Heap: GATE CSE 2006 | Question: 77 top

☛ https://gateoverflow.in/87191

✔



Heap will be constructed like this, based on the correct answer of Q.76 $(\text{which is} \quad 9 \quad 5 \quad 6 \quad 8 \quad 3 \quad 1)$

Correct Answer: A

15 votes -- Anurag Pandey (10.5k points)

3.10.11 Heap: GATE CSE 2007 | Question: 47 top

https://gateoverflow.in/1245

✔ Number of elements in the path from new leaf to root $= \log n$, and all elements are sorted from leaf to root so we can do a binary search which will result in $O(\log \log n)$ number of comparisons.

Since a heap is a complete binary tree (hence height balanced also), in an array implementation, from every node index, we can know its depth and this will be the number of elements $- n$ for binary search.

Correct Answer: B

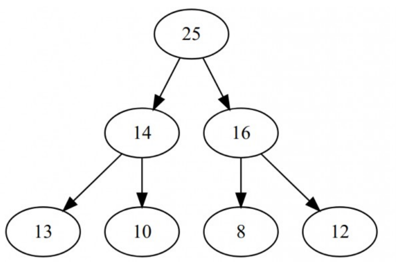
94 votes -- Vikrant Singh (11.2k points)

3.10.12 Heap: GATE CSE 2009 | Question: 59 top

https://gateoverflow.in/1341

✔ **Answer : (C)**

The binary max-Heap looks like this :



Max-heap

13 votes -- Dipak Majhi (757 points)

Taking the given array as level order traversal, we can build binary tree.

(A) 13 comes as child of 12, which is not allowed in a binary max-heap

(B) 16 comes as child of 14 violating max-heap property

(C) is a valid binary max-heap as all children are smaller than their parent

(D) 16 comes as child of 12, violating max-heap property

18 votes -- Arjun Suresh (330k points)

3.10.13 Heap: GATE CSE 2009 | Question: 60 top

https://gateoverflow.in/43466

✔ During delete, the root element is removed, replaced with the last element and heap property is corrected by pushing the root downwards. So, for first delete, 25 14 16 13 10 8 12 $\rightarrow$ 12 14 16 13 10 8 $\rightarrow$ 16 14 12 13 10 8 (the element not satisfying max-heap property is exchanged with the largest of its children) (heap property satisfied)

Second delete:

16 14 12 13 10 8 $\rightarrow$ 8 14 12 13 10 $\rightarrow$ 14 8 12 13 10 $\rightarrow$ 14 13 12 8 10 (heap property satisfied)

Correct Answer: D

29 votes -- Arjun Suresh (330k points)

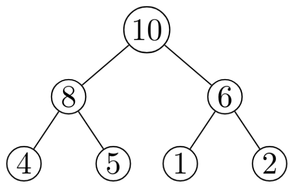
3.10.14 Heap: GATE CSE 2011 | Question: 23 top

https://gateoverflow.in/2125

✔ In option (A) - it is not a max heap because it is not a Complete Binary Tree (a heap must have all levels till last but one completely filled and in the last level all nodes must be filled from the left end without a gap till the last node)

In option (C) - it is complete binary tree but is not following the max heap property i.e. the value of parent node is not always greater than the child nodes as the node of value 5 is less then one of its child node value of 8.

In option (D) - similar to (C) option explanation here node of value 2 is less than the child node value 4.



Correct option is (**B**) and it satisfies all the properties of a max heap.

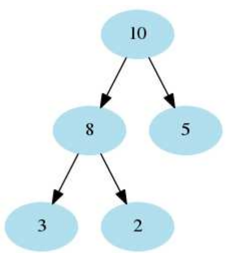
22 votes -- ASHU2015 (261 points)

3.10.15 Heap: GATE CSE 2014 Set 2 | Question: 12 top

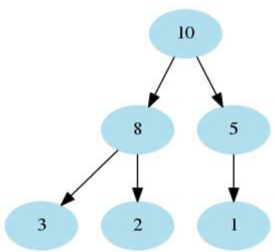
https://gateoverflow.in/1967

✔ Answer is (**A**)....whenever insertion will be done in heap ,it will always inserted in last level from left to right.so we insert 1 and 7 as a child of node 5 now we perform heapify algorithm until heap property will satisfied..and then we get the heap whose level order traversal is $10, 8, 7, 3, 2, 1, 5$.

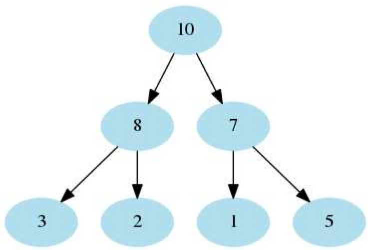
Initial heap



After insert of 1



After insert of 7



23 votes -- neha pawar (3.3k points)

3.10.16 Heap: GATE CSE 2015 Set 1 | Question: 32 top https://gateoverflow.in/8273

Heap is complete binary tree. To insert a new element, we put it at the end of the tree and move up towards root till heap property is satisfied. Here, 35 comes as child of 15, with the path $40-30-15-35$. So, we swap $15, 35$ and then $30, 35$ to get the new path $40-35-30-15$. So, new heap will be $40\ 35\ 20\ 10\ 30\ 16\ 17\ 8\ 4\ 15$.

Correct Answer: B

39 votes -- Arjun Suresh (330k points)

3.10.17 Heap: GATE CSE 2015 Set 2 | Question: 17 top https://gateoverflow.in/8091

Answer is (**A**).

Here, lower bound imply best algorithm which works for all cases and hence we should consider worst-case.

Max-Heapify(root).

52 votes -- Vikrant Singh (11.2k points)

3.10.18 Heap: GATE CSE 2015 Set 3 | Question: 19 top https://gateoverflow.in/8418

First interchange 15-100
2^{nd} 50-100
3^{rd} 89-100
So, total interchange 3 so option (**D**) is correct.

30 votes -- Anoop Sonkar (4.1k points)

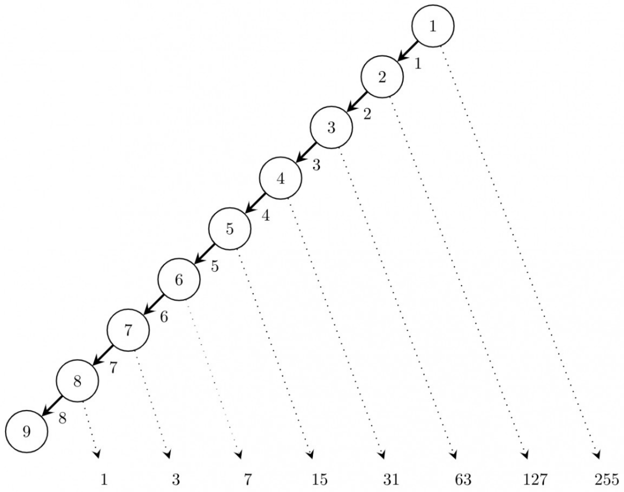
3.10.19 Heap: GATE CSE 2016 Set 1 | Question: 37 top https://gateoverflow.in/39706

Answer would be (**B**) $O(d)$ but not $O(1)$.. as we need to apply heapify.. and suppose if we are deleting root, in worst case would take $O(d)$ time..

52 votes -- Abhilash Panicker (7.6k points)

3.10.20 Heap: GATE CSE 2016 Set 2 | Question: 34 top https://gateoverflow.in/39585

Here answer is 8. With 1023 nodes, we can easily build a min heap as shown in the below diagram.



Here, 9 is pushed to the deepest possible level which is 8 here. (k^{th} smallest element in a min-heap cannot go deeper than level k because the path from root to that level goes through $k-1$ smaller elements). Now, do we have enough elements to fill the right subtree to satisfy the complete binary tree property required by a heap? Yes. As shown in the diagram, up to depth 9 (assume it is fully filled) we'll need $2^9 - 1 = 511$ elements. We have 1023 elements and so the remaining 512 elements should go to depth 9 which is guaranteed to have the maximum element – here 1023.

Correct answer is 8.

81 votes -- Akash Kanase (36k points)

3.10.21 Heap: GATE CSE 2019 | Question: 40 top

https://gateoverflow.in/302808

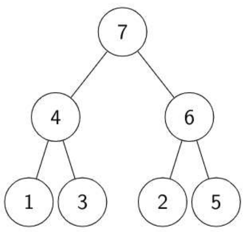
✔

I. The smallest element in a max-heap is always at a leaf node – TRUE because by definition of a max-heap every parent node must be larger than its child node.
II. The second largest element in a max-heap is always a child of a root node – TRUE. The k^{th} largest element cannot have more than $k-1$ parents (larger value nodes) and so cannot be lower than level k.
III. A max-heap can be constructed from a binary search tree in $\Theta(n)$ time. Build heap for any array of size n (even unsorted) can be done in $O(n)$ time.

Now lets see the 4^{th} statement.

4. A binary search tree can be constructed from a max-heap in $\Theta(n)$ time. This can be proved $\textsf{FALSE}$ using the simple argument that we can do max-heap construction in $O(n)$ and if we can convert this to $\textsf{BST}$ in $O(n)$ time we can do an inorder traversal of $\textsf{BST}$ in $O(n)$ time and thus we manage to sort n elements in $O(n)$ time using just comparisons which is known to take at least $\Omega(n \log n)$ time.

An example construction of $\textsf{BST}$ from a max-heap.



Max – Heap

Max Heap Array Representation:

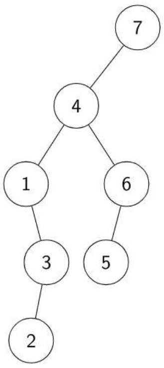
Value	7	4	6	1	3	2	5
Array Index	1	2	3	4	5	6	7

Make binary search tree using Array Representation:

1. Pick elements from $A[1]$ to $A[7]$ one by one.

2. Compare with all previous elements and find its respective place.

We'll get Binary Search Tree as follows:



Number of comparisons in the worst case for each element $= \overset{\text{Elements}}{\underbrace{\begin{matrix}1 & 2 & 3 & 4 & 5 & 6 & 7\\ 0 & 1 & 2 & 3 & 4 & 5 & 6\end{matrix}}_{\textbf{Comparisons}}}$

So for n element heap the total no of comparisons could be
$= 0 + 1 + 2 + \dots (n-2) + (n-1)$
$= \Theta \frac{(n-1)n}{2}$
$= \Theta(n^2)$
Note: By using more efficient method this time complexity could be reduced to $O(n. \log n)$ but it can never be $O(n)$.
OPTION A is the correct answer.

11 votes -- Nitesh Singh (4.9k points)

3.10.22 Heap: GATE IT 2004 | Question: 53 top https://gateoverflow.in/3696

✔ By using **Build Heap method** we can create heap from complete binary tree.
which will take $O(n)$.
Correct Answer: B

41 votes -- Sneha Goel (819 points)

3.10.23 Heap: GATE IT 2006 | Question: 44 top https://gateoverflow.in/3587

✔ For a heap(max heap) parent should be greater than or equal to children. in a heap of $[1..n]$ left child of ith node will be at $2 * i$ th position and right child will be at $2 * i + 1$ position.

So, for given options we can verify it.

.
Option **C** seems to be following the property.

16 votes -- Sankaranarayanan P.N (8.5k points)

3.10.24 Heap: GATE IT 2006 | Question: 72 top https://gateoverflow.in/3616

✔ Here we need to call Heapify/ Bubble down procedure on Root. Which in worst case will take time O (log n). So B is correct option.

Other options does not even make sense , because with O(n) we can even build entire Heap not just heapify on root. O (n log n) & O (n log log n) is more than O(n)

23 votes -- Akash Kanase (36k points)

3.11 Infix Prefix (3) top

3.11.1 Infix Prefix: GATE CSE 1989 | Question: 4-ii top https://gateoverflow.in/87881

Compute the postfix equivalent of the following infix arithmetic expression

$a + b * c + d * e \uparrow f$

where $\uparrow$ represents exponentiation. Assume normal operator precedences.

gate1989 descriptive data-structures stack infix-prefix

Answer

3.11.2 Infix Prefix: GATE CSE 1997 | Question: 1.7 top

https://gateoverflow.in/2223

Which of the following is essential for converting an infix expression to the postfix form efficiently?

A. An operator stack
B. An operand stack
C. An operand stack and an operator stack
D. A parse tree

gate1997 normal infix-prefix stack data-structures

Answer

3.11.3 Infix Prefix: GATE CSE 1998 | Question: 19b top

https://gateoverflow.in/15708

Compute the post fix equivalent of the following expression $3^* \log(x+1) - \frac{a}{2}$

gate1998 stack infix-prefix descriptive

Answer

Answers: Infix Prefix

3.11.1 Infix Prefix: GATE CSE 1989 | Question: 4-ii top

https://gateoverflow.in/87881

✔ $= a + (bc*) + (d(ef \uparrow)*)$

$= (abc * +) + (def \uparrow *)$

$= abc * + def \uparrow * +$

9 votes

-- Aboveallplayer (12.5k points)

3.11.2 Infix Prefix: GATE CSE 1997 | Question: 1.7 top

https://gateoverflow.in/2223

✔ A. **An operator stack** // **Infix to (Postfix or Prefix)**

B. An operand stack //Postfix or Prefix Evaluation

C. An operand stack and an operator stack //we never use two stacks

But for Prefix to (Infix or postfix) OR Postfix to (Infix or prefix) We can use a stack where both operator and operand can present simultaneously

D. A parse tree // Not relevant to this question

Hence, **Option A** is Answer.

http://condor.depaul.edu/ichu/csc415/notes/notes9/Infix.htm is a good read.

References

33 votes

-- Rajesh Pradhan (18.9k points)

3.11.3 Infix Prefix: GATE CSE 1998 | Question: 19b top

https://gateoverflow.in/15708

✔ According to http://faculty.washington.edu/jstraub/dsa/aexp/

The priority of the operators follows the usual conventions:

- The highest priority is assigned to unary operators (note that, in this context, a function such as *sin* is considered a unary operator). All unary operators have the same priority.
- Exponentiation has the second highest priority.
- The third highest priority is assigned to the multiplication and division operators.
- The lowest priority is given to the addition and subtraction operators.

Example:->>
Infix expression: $3 * \log(10)$
Postfix expression:
$= 3 * (10 \log)$ //(Priority of unary operator *log* forces $\log(10)$ to evaluate first.)
$= 3\ 10 \log *$

Now for our case $3 * \log(x+1) - a/2$

First content inside parenthesis will be evaluated

So, $x+1$ will become $x\ 1+$

Now among $(*, /, \log, +, -)$ operators, log has highest priority as it is the only unary operator

So, $\log(x\ 1+)$ will become $x\ 1 + \log$

Now suppose $z = x1 + \log$ and we get $3 * z - a/2$

$\implies 3z * a\ 2/-$

Now, substitute $z = x\ 1 + \log$ and we get

3x 1 $+ \log$ ***a 2**$/-$ as answer.

References

👍 81 votes -- Rajesh Pradhan (18.9k points)

3.12 Linked Lists (20) top

3.12.1 Linked Lists: GATE CSE 1987 | Question: 1-xv top https://gateoverflow.in/80298

In a circular linked list oraganisation, insertion of a record involves modification of

A. One pointer.
B. Two pointers.
C. Multiple pointers.
D. No pointer.

gate1987 data-structures linked-lists

Answer

3.12.2 Linked Lists: GATE CSE 1987 | Question: 6a top https://gateoverflow.in/82419

A list of n elements is commonly written as a sequence of n elements enclosed in a pair of square brackets. For example. $[10, 20, 30]$ is a list of three elements and $[]$ is a nil list. Five functions are defined below:

- $car(l)$ returns the first element of its argument list l;
- $cdr(l)$ returns the list obtained by removing the first element of the argument list l;
- $glue(a, l)$ returns a list m such that $car(m) = a$ and $cdr(m) = l$.
- $f(x, y) \equiv$ if $x = []$ then y
  else $glue(car(x), f(cdr(x), y))$;
- $g(x) \equiv$ if $x = []$ then $[]$
  else $f(g(cdr(x)), glue(car(x), []))$

What do the following compute?

a. $f([32, 16, 8], [9, 11, 12])$
b. $g([5, 1, 8, 9])$

gate1987 data-structures linked-lists descriptive

Answer

3.12.3 Linked Lists: GATE CSE 1993 | Question: 13 top

https://gateoverflow.in/2310

Consider a singly linked list having n nodes. The data items $d_1, d_2, \dots d_n$ are stored in these n nodes. Let X be a pointer to the j^{th} node $(1 \leq j \leq n)$ in which d_j is stored. A new data item d stored in node with address Y is to be inserted. Give an algorithm to insert d into the list to obtain a list having items $d_1, d_2, \dots, d_j, d, \dots, d_n$ in order without using the header.

gate1993 data-structures linked-lists normal descriptive

Answer

3.12.4 Linked Lists: GATE CSE 1994 | Question: 1.17, UGCNET-Sep2013-II: 32 top

https://gateoverflow.in/2460

Linked lists are not suitable data structures for which one of the following problems?

A. Insertion sort
B. Binary search
C. Radix sort
D. Polynomial manipulation

gate1994 data-structures linked-lists normal ugcnetsep2013ii

Answer

3.12.5 Linked Lists: GATE CSE 1995 | Question: 2.22 top

https://gateoverflow.in/2634

Which of the following statements is true?

I. As the number of entries in a hash table increases, the number of collisions increases.
II. Recursive programs are efficient
III. The worst case complexity for Quicksort is $O(n^2)$
IV. Binary search using a linear linked list is efficient

A. I and II
B. II and III
C. I and IV
D. I and III

gate1995 data-structures linked-lists hashing

Answer

3.12.6 Linked Lists: GATE CSE 1997 | Question: 1.4 top

https://gateoverflow.in/2220

The concatenation of two lists is to be performed on $O(1)$ time. Which of the following implementations of a list should be used?

A. Singly linked list
B. Doubly linked list
C. Circular doubly linked list
D. Array implementation of list

gate1997 data-structures linked-lists easy

Answer

3.12.7 Linked Lists: GATE CSE 1997 | Question: 18 top

https://gateoverflow.in/2278

Consider the following piece of 'C' code fragment that removes duplicates from an ordered list of integers.

```
Node *remove-duplicates (Node* head, int *j)
{
    Node *t1, *t2; *j=0;
    t1 = head;
    if (t1! = NULL)
        t2 = t1 ->next;
    else return head;
    *j = 1;
    if(t2 == NULL) return head;
    while (t2 != NULL)
    {
        if (t1.val != t2.val) ----------------> (S1)
        {
            (*j)++;
            t1 -> next = t2;
            t1 = t2; -----> (S2)
        }
            t2 = t2 ->next;
    }
    t1 -> next = NULL;
    return head;
}
```

Assume the list contains n elements ($n \geq 2$) in the following questions.

a. How many times is the comparison in statement $S1$ made?
b. What is the minimum and the maximum number of times statements marked $S2$ get executed?
c. What is the significance of the value in the integer pointed to by j when the function completes?

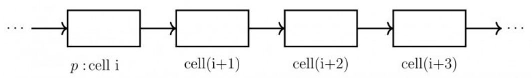
gate1997 data-structures linked-lists normal descriptive

Answer

3.12.8 Linked Lists: GATE CSE 1998 | Question: 19a top

https://gateoverflow.in/1733

Let p be a pointer as shown in the figure in a single linked list.



What do the following assignment statements achieve?

```
q:= p -> next
p -> next:= q -> next
q -> next:=(q -> next) -> next
(p -> next) -> next:= q
```

gate1998 data-structures linked-lists normal descriptive

Answer

3.12.9 Linked Lists: GATE CSE 1999 | Question: 11b top

https://gateoverflow.in/93575

Write a constant time algorithm to insert a node with data D just before the node with address p of a singly linked list.

gate1999 data-structures linked-lists descriptive

Answer

3.12.10 Linked Lists: GATE CSE 2002 | Question: 1.5 top

https://gateoverflow.in/809

In the worst case, the number of comparisons needed to search a single linked list of length n for a given element is

A. $\log n$
B. $\frac{n}{2}$
C. $\log_2 n - 1$
D. n

gate2002-cse easy data-structures linked-lists

Answer

3.12.11 Linked Lists: GATE CSE 2003 | Question: 90 top

☛ https://gateoverflow.in/973

Consider the function f defined below.

```
struct item {
        int data;
        struct item * next;
};
int f(struct item *p) {
    return ((p == NULL) || (p->next == NULL)||
        ((p->data <= p ->next -> data) &&
        f(p->next)));
}
```

For a given linked list p, the function f returns 1 if and only if

A. the list is empty or has exactly one element
B. the elements in the list are sorted in non-decreasing order of data value
C. the elements in the list are sorted in non-increasing order of data value
D. not all elements in the list have the same data value

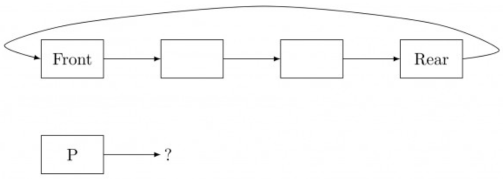
gate2003-cse data-structures linked-lists normal

Answer

3.12.12 Linked Lists: GATE CSE 2004 | Question: 36 top

☛ https://gateoverflow.in/1033

A circularly linked list is used to represent a Queue. A single variable p is used to access the Queue. To which node should p point such that both the operations enQueue and deQueue can be performed in constant time?



A. rear node
B. front node
C. not possible with a single pointer
D. node next to front

gate2004-cse data-structures linked-lists normal

Answer

3.12.13 Linked Lists: GATE CSE 2004 | Question: 40 top

☛ https://gateoverflow.in/1037

Suppose each set is represented as a linked list with elements in arbitrary order. Which of the operations among $\text{union, intersection, membership, cardinality}$ will be the slowest?

A. union only
B. $\text{intersection, membership}$
C. $\text{membership, cardinality}$
D. $\text{union, intersection}$

gate2004-cse data-structures linked-lists normal

Answer

3.12.14 Linked Lists: GATE CSE 2008 | Question: 62 top

☛ https://gateoverflow.in/485

The following C function takes a single-linked list of integers as a parameter and rearranges the elements of the list. The function is called with the list containing the integers $1, 2, 3, 4, 5, 6, 7$ in the given order. What will be the contents of the list after function completes execution?

```
struct node {
    int value;
    struct node *next;
};

void rearrange(struct node *list) {
    struct node *p, *q;
    int temp;
    if (!list || !list -> next) return;
    p = list; q = list -> next;
    while(q) {
        temp = p -> value; p->value =  q -> value;
        q->value = temp; p = q ->next;
        q = p? p ->next : 0;
    }
}
```

A. $1, 2, 3, 4, 5, 6, 7$
B. $2, 1, 4, 3, 6, 5, 7$
C. $1, 3, 2, 5, 4, 7, 6$
D. $2, 3, 4, 5, 6, 7, 1$

gate2008-cse data-structures linked-lists normal

Answer

3.12.15 Linked Lists: GATE CSE 2010 | Question: 36 top

https://gateoverflow.in/2337

The following C function takes a singly-linked list as input argument. It modifies the list by moving the last element to the front of the list and returns the modified list. Some part of the code is left blank.

```
typedef struct node
{
    int value;
    struct node *next;
} node;
Node *move_to-front(Node *head)
{
    Node *p, *q;
    if ((head == NULL) || (head -> next == NULL))
        return head;
    q = NULL;
    p = head;
    while (p->next != NULL)
    {
        q=p;
        p=p->next;
    }
    _______________

    return head;

}
```

Choose the correct alternative to replace the blank line.

A. $q = NULL; p \rightarrow next = head; head = p$;
B. $q \rightarrow next = NULL; head = p; p \rightarrow next = head$;
C. $head = p; p \rightarrow next = q; q \rightarrow next = NULL$;
D. $q \rightarrow next = NULL; p \rightarrow next = head; head = p$;

gate2010-cse data-structures linked-lists normal

Answer

3.12.16 Linked Lists: GATE CSE 2016 Set 2 | Question: 15 top

https://gateoverflow.in/39557

N items are stored in a sorted doubly linked list. For a *delete* operation, a pointer is provided to the record to be deleted. For a *decrease-key* operation, a pointer is provided to the record on which the operation is to be performed.

An algorithm performs the following operations on the list in this order: $\Theta(N)$ *delete*, $O(\log N)$ *insert*, $O(\log N)$ *find*, and $\Theta(N)$ *decrease-key*. What is the time complexity of all these operations put together?

A. $O(\log^2 N)$
B. $O(N)$

C. $O(N^2)$
D. $\Theta\left(N^2 \log N\right)$

gate2016-cse-set2 data-structures linked-lists time-complexity normal

Answer

3.12.17 Linked Lists: GATE CSE 2017 Set 1 | Question: 08 top https://gateoverflow.in/118711

Consider the C code fragment given below.

```
typedef struct node {
    int data;
    node* next;
} node;

void join(node* m, node* n) {
    node* p = n;
    while(p->next != NULL) {
        p = p->next;
    }
    p->next = m;
}
```

Assuming that m and n point to valid NULL-terminated linked lists, invocation of join will

A. append list m to the end of list n for all inputs.
B. either cause a null pointer dereference or append list m to the end of list n.
C. cause a null pointer dereference for all inputs.
D. append list n to the end of list m for all inputs.

gate2017-cse-set1 data-structures linked-lists normal

Answer

3.12.18 Linked Lists: GATE CSE 2020 | Question: 16 top https://gateoverflow.in/333215

What is the worst case time complexity of inserting n elements into an empty linked list, if the linked list needs to be maintained in sorted order?

A. $\Theta(n)$
B. $\Theta(n \log n)$
C. $\Theta(n)^2$
D. $\Theta(1)$

gate2020-cse linked-lists

Answer

3.12.19 Linked Lists: GATE IT 2004 | Question: 13 top https://gateoverflow.in/3654

Let P be a singly linked list. Let Q be the pointer to an intermediate node x in the list. What is the worst-case time complexity of the best-known algorithm to delete the node x from the list ?

A. $O(n)$
B. $O(\log^2 n)$
C. $O(\log n)$
D. $O(1)$

gate2004-it data-structures linked-lists normal ambiguous

Answer

3.12.20 Linked Lists: GATE IT 2005 | Question: 54 top https://gateoverflow.in/3815

The following C function takes a singly-linked list of integers as a parameter and rearranges the elements of the list. The list is represented as pointer to a structure. The function is called with the list containing the integers $1, 2, 3, 4, 5, 6, 7$ in the given order. What will be the contents of the list after the function completes execution?

```
struct node {int value; struct node *next;};
void rearrange (struct node *list) {
    struct node *p, *q;
    int temp;
    if (!list || !list -> next) return;
    p = list; q = list -> next;
    while (q) {
        temp = p -> value;
        p -> value = q -> value;
        q -> value = temp;
        p = q -> next;
        q = p ? p -> next : 0;
      }
}
```

A. $1, 2, 3, 4, 5, 6, 7$
B. $2, 1, 4, 3, 6, 5, 7$
C. $1, 3, 2, 5, 4, 7, 6$
D. $2, 3, 4, 5, 6, 7, 1$

gate2005-it data-structures linked-lists normal

Answer

Answers: Linked Lists

3.12.1 Linked Lists: GATE CSE 1987 | Question: 1-xv top

https://gateoverflow.in/80298

✔ Suppose we have to insert node p after node q then

- $p \rightarrow \text{next} = q \rightarrow \text{next}$
- $q \rightarrow \text{next} = p$

So, two pointers,

Answer is (**B**).

20 votes

-- Sanket_ (3.1k points)

3.12.2 Linked Lists: GATE CSE 1987 | Question: 6a top

https://gateoverflow.in/82419

✔ Part (a) Concatenate the two lists.

Part (b) Reverse the given list.

Part (a):

$$f([32, 16, 8], [9, 11, 12])$$
$$\Downarrow$$
$$\text{glue}(\text{car}[32, 16, 8], f(\text{cdr}([32, 16, 8]), [9, 11, 12]))$$
$$\Downarrow$$
$$\text{glue}(32, f([16, 8], [9, 11, 12]))$$
$$\Downarrow$$
$$\text{glue}(32, \text{glue}(\text{car}([16, 8]), f(\text{cdr}([16, 8]), [9, 11, 12])))$$
$$\Downarrow$$
$$\text{glue}(32, \text{glue}(16, f([8], [9, 11, 12])))$$
$$\Downarrow$$
$$\text{glue}(32, \text{glue}(16, \text{glue}(\text{car}([8]), f(\text{cdr}([8]), [9, 11, 12]))))$$
$$\Downarrow$$
$$\text{glue}(32, \text{glue}(16, \text{glue}(8, f([]), [9, 11, 12])))$$
$$\Downarrow$$
$$\text{glue}(32, \text{glue}(16, \text{glue}(8, [9, 11, 12])))$$
$$\Downarrow$$
$$\text{glue}(32, \text{glue}(16, [8, 9, 11, 12]))$$

$\Downarrow$

$\text{glue}(32, [16, 8, 9, 11, 12])$

$\Downarrow$

$\boxed{[32, 16, 8, 9, 11, 12]}$ **Answer.**

Part (b):

$g([5, 1, 8, 9])$

$\Downarrow$

$f(g(\text{cdr}([5, 1, 8, 9])), \text{glue}(\text{car}([5, 1, 8, 9]), []))$

$\Downarrow$

$f(g([1, 8, 9]), \text{glue}(5, [])))$

$\Downarrow$

$f(g([1, 8, 9]), [5])$

So, we can say, $g([5, 1, 8, 9]) = f(g([1, 8, 9]), [5])$

Similarly,

$g([1, 8, 9]) = f(g([8, 9]), [1])$

$g([8, 9]) = f(g([9]), [8])$

$g([9]) = f(g([]), [9]) = f([], [9]) = [9]$

Now, backtrack

$g([8, 9]) = f([9], [8]) = [9, 8]$ (From part (a))

& $g([1, 8, 9]) = f([9, 8], [1])$

$g([1, 8, 9]) = [9, 8, 1]$ (From part (a))

Now, $g([5, 1, 8, 9]) = f([9, 8, 1], [5]) = [9, 8, 1, 5]$

So, $\boxed{g([5, 1, 8, 9]) = [9, 8, 1, 5]}$ **Answer**

11 votes -- ankitgupta.1729 (15k points)

3.12.3 Linked Lists: GATE CSE 1993 | Question: 13 top https://gateoverflow.in/2310

✔ Following 2 lines are enough to realize above constraint :

```
1. Y->next = X-> next
2. X->next = Y
```

22 votes -- Rajesh Pradhan (18.9k points)

3.12.4 Linked Lists: GATE CSE 1994 | Question: 1.17, UGCNET-Sep2013-II: 32 top https://gateoverflow.in/2460

✔ **Linked lists are suitable for:**

Insertion sort: No need to swap here just find appropriate place and join the link

Polynomial manipulation: Linked List is a natural solution for polynomial manipulation

Radix sort: Here we are putting digits according to same position(unit,tens) into buckets; which can be effectively handled by linked lists.

Not Suitable for:

Binary search: Because finding mid element itself takes $O(n)$ time.

So, Option **B** is answer.

32 votes -- Rajesh Pradhan (18.9k points)

3.12.5 Linked Lists: GATE CSE 1995 | Question: 2.22 top https://gateoverflow.in/2634

✔ Correct Option: D

Binary search using a linked list is not efficient as it will not give $O(\log n)$, because we will not be able to find the mid in constant time. Finding mid in linked list takes $O(n)$ time.

Recursive programs are not efficient because they take a lot of space, Recursive methods will often throw Stack Overflow Exception while processing big sets. moreover, it has its own advantages too.

31 votes -- Gate Keeda (15.9k points)

3.12.6 Linked Lists: GATE CSE 1997 | Question: 1.4 top https://gateoverflow.in/2220

✔ **Option C (Circular Doubly linked List)**

Analyze below Code which is $O(1)$

Suppose List1's first element is pointed by pointer $p1$

And List2's first element is pointed by $p2$

And tmp is a temporary pointer of node type.

```
p1->prev->next = p2 ;

tmp= p2-> prev ;

p2-> prev= p1-> prev ;

p1-> prev = tmp;

tmp -> next = p1;
```

Options A&B of linked list are not possible in $O(1)$. Because they cannot find out rear element without doing a linear traversal.

For **Option D.** Array implementation requires $O(n_1 + n_2)$ copy operations where n_1 and n_2 represent the sizes of List1 and List2 respectively.

30 votes -- Rajesh Pradhan (18.9k points)

3.12.7 Linked Lists: GATE CSE 1997 | Question: 18 top https://gateoverflow.in/2278

✔

a. As we are comparing here pair wise so for n elements we require compulsory $n-1$ comparison
b. $S2$ is executed only for distinct elements so **max** $n-1$ times and **min** 0 when all r duplicates or list contain no or 1 element.
c. j holds the count on number of distinct elements in the ordered list.

27 votes -- Rajesh Pradhan (18.9k points)

3.12.8 Linked Lists: GATE CSE 1998 | Question: 19a top https://gateoverflow.in/1733

✔ Swaps the two nodes next to p in the linked list.

33 votes -- Arjun Suresh (330k points)

3.12.9 Linked Lists: GATE CSE 1999 | Question: 11b top https://gateoverflow.in/93575

✔ Let A, B, C, D, E, F be the data.
$A \to B \to C \to E \to F$

Let the pointer to E be p.

A node with data D has to be inserted before E.
I'll do one thing - add D just after node E and it'll take constant time.

$p \to next = D_{addr}$
$D_{addr} \to next = F_{addr}$

$A \to B \to C \to E \to D \to F$

Take a temporary variable and swap the values E and D.

```
temp = p-> data
p->data = p->next->data
p->next->data = temp
```

Now linked list will look like
$A \to B \to C \to D \to E \to F$

Still one more work left - now pointer p pointing to D so increment pointer p to point data E.
$p = p \rightarrow next$

```
void InsertBeforeGivenPointer(struct node* p, int D){
    struct node* temp = (node*)malloc(sizeof(struct node));
    temp->next = p->next;
    p->next = temp;
    temp->data = p->data;
    p->data = D;
}
```

40 votes -- Digvijay (44.9k points)

3.12.10 Linked Lists: GATE CSE 2002 | Question: 1.5 top https://gateoverflow.in/809

✔ **A & C** are not correct as we can not do binary search in Linked list.

B seems like average case, be we are asked for worst case.

Worst case is we do not find the element in list. We might end up searching entire list & comparing with each element. So, answer -> (**D**). n

24 votes -- Akash Kanase (36k points)

3.12.11 Linked Lists: GATE CSE 2003 | Question: 90 top https://gateoverflow.in/973

✔ It returns 1 if and only if the linked list is sorted in non-decreasing order- **(B)** option.

It returns 1 if the list is empty or has just 1 element also, but if and only if makes (A) option false.

34 votes -- Bhagirathi Nayak (11.7k points)

3.12.12 Linked Lists: GATE CSE 2004 | Question: 36 top https://gateoverflow.in/1033

✔ **The pointer points to the Rear node.**

EnQueue: Insert newNode after Rear, and make Rear point to the newly inserted node:

```
//struct node *newNode;
newNode->next = rear->next;
rear->next = newNode;
rear=newNode;
```

DeQueue: Delete the Front node, and make the second node the front node.

```
//rear->next points to the front node.
//front->next points to the second node.
struct node* front;
front = rear->next;
rear->next = front->next;
free(front);
```

92 votes -- Pragy Agarwal (18.3k points)

3.12.13 Linked Lists: GATE CSE 2004 | Question: 40 top https://gateoverflow.in/1037

✔ Answer is (**D**).

Membership is linear search - $O(n_1 + n_2)$

Cardinality is linear - $O(n_1 + n_2)$

For union we need to ensure no duplicate elements should be present - $O(n_1 \times n_2)$ for each element we need to check if that element exists in other set

For intersection also for every element in set1 we need to scan set2 - $O(n_1 \times n_2)$

74 votes -- Ankit Rokde (6.9k points)

3.12.14 Linked Lists: GATE CSE 2008 | Question: 62 top

☛ https://gateoverflow.in/485

✔ The loop is interchanging the adjacent elements of the list. But after each interchange, next interchange starts from the unchanged elements only (due to $p = q \rightarrow next$;).

1st iteration $1, 2, 3, 4, 5, 6, 7$
$\Rightarrow 2, 1, 3, 4, 5, 6, 7$

2nd iteration $2, 1, 4, 3, 5, 6, 7$

3rd iteration $2, 1, 4, 3, 6, 5, 7$

p pointing to null q pointing to 7, as p is false hence $q = p?\ p \rightarrow next$:0; will return $q = 0$ ending the loop

Answer is **option B.**

👍 30 votes -- Manali (2.1k points)

3.12.15 Linked Lists: GATE CSE 2010 | Question: 36 top

☛ https://gateoverflow.in/2337

✔ As per given code p points to last node which should be head in modified.

q is the previous node of tail which should be tail for modified

Answer is (**D**).

👍 37 votes -- Sankaranarayanan P.N (8.5k points)

3.12.16 Linked Lists: GATE CSE 2016 Set 2 | Question: 15 top

☛ https://gateoverflow.in/39557

✔ Answer is (**C**).

- Delete $O(1)$
- Insert $O(N)$
- Find $O(N)$

Decrease Key $\Rightarrow O(N)$ //Because we need to search position in Linked list. (It is similar to a Delete followed by an Insert with the decreased value)

- $O(n)$ delete $\Rightarrow O(N \times 1) = O(N)$
- $O(\log N)$ find $\Rightarrow O(\log N \times N) = O(N \log N)$
- $O(\log N)$ insert $\Rightarrow O(N \log N)$
- $O(N)$ decrease key $\Rightarrow O(N \times N) = O(N^2)$

Even though it says at start we got N elements, then we are deleting $Q(N)$ elements, here $Q(N)$ can be anything like $N/2, N/4, N/3$ and list need not be empty, then above explanation holds !

In case it actually deleted all elements at start analysis will be something like below $\Rightarrow$

All N elements are deleted, Time complexity $O(1)$ each delete, total delete $O(N)$

Now $\log N$ insert, it'll take $1 + 2 + \log N$ time, then $(\log N * \log N - 1)/2 \Rightarrow O((\log N)^2)$

Now $\log N$ finds $\Rightarrow$ it'll take $\log N$ time per find (because find take $O(N)$ but here $N = \log N$)
$\Rightarrow O((\log N)^2)$

Now N decrease key, Size of list is $\log N$, each decrease key can take $O(size\ of\ list)$

So, size of list * no. of decrease key $\Rightarrow N \times \log N = O(N \log N)$

There is no option like $O(N \log N)$

So, correct answer is $O(N^2)$.

👍 107 votes -- Akash Kanase (36k points)

3.12.17 Linked Lists: GATE CSE 2017 Set 1 | Question: 08 top

☛ https://gateoverflow.in/118711

✔ Here is the implemented code in c (-std=c99).

```
#include <stdio.h>
```

```
#include <stdlib.h>

#define M 5
#define N 4

int M_data[M] = {1,2,3,4,5};
int N_data[N] = {6,7,8,9};

typedef struct node {
int data;
struct node * next;
}node;

void join(node *m,node *n) {
    node * p = n;
    while(p->next != NULL) p = p->next;
    p->next = m;
}

node * bulk_insert(int list_data[],int size) {
    node * head = NULL;
    for(int i=0;i<size;i++) {
        node * newnode = malloc(sizeof(node));
        newnode->data = list_data[i];
        newnode->next = NULL;

        if(head == NULL) {
            head = newnode;
        }else {
            node * temp = head;
            while(temp->next != NULL) temp = temp->next;
            temp->next = newnode;
        }
    }
    return head;
}
void display(node *);
void list_dealloc(node *); /*deallocation prototype*/

int main() {
    node * m = NULL;
    node * n = NULL;
    // insert m_list data
    m = bulk_insert(M_data,M);
    n = bulk_insert(N_data,N); // commenting this causes runtime error
    // is list n is empty
    printf("\n before join operation :\n");
    display(m);
    display(n);

    join(m,n);

    printf("\n after join operation :\n");
    display(m);
    display(n);

    //list_dealloc(m); no need now
    list_dealloc(n); // OK
    return 0;

}

void display(node *head) {
    while(head != NULL) {
        printf("%d->",head->data);
        head = head->next;
    }
    printf("null\n");
}
void list_dealloc(node * head) {
    while(head != NULL) {
        node * temp = head;
        head = head->next;
        free(temp);
    }
}
```

With both **n** and **m** and n being **non-empty** linked list, then,

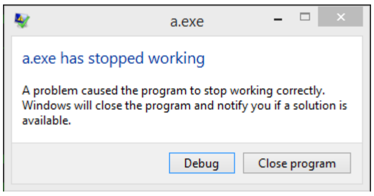
O/P:

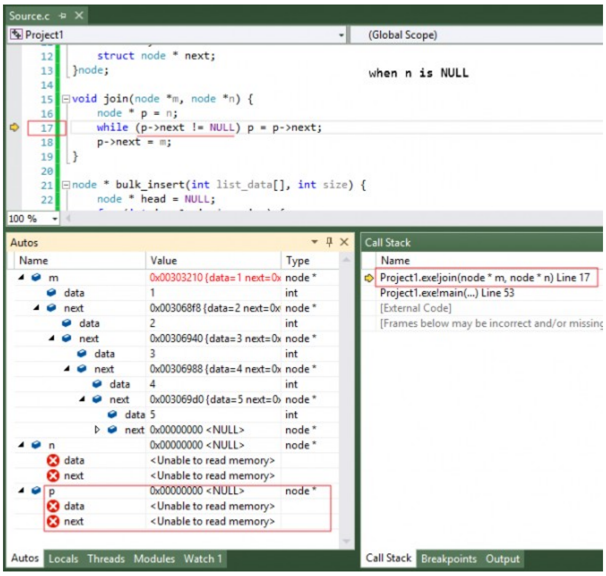
```
before join operation :
```

```
1->2->3->4->5->null
6->7->8->9->null

 after join operation :
1->2->3->4->5->null
6->7->8->9->1->2->3->4->5->null
```

With n being **empty** linked list, then,

O/P:





Correct Answer: B

👍 52 votes -- Debashish Deka (40.7k points)

3.12.18 Linked Lists: GATE CSE 2020 | Question: 16 top

☛ https://gateoverflow.in/333215

✔ "Inserting n elements into an empty linked list, list needs to be maintained in sorted order".

It is not mentioned if the "n elements" are given initially itself or we must insert one element at a time.

1. For the former case, we can just sort the initial n element array - can be done in $O(n \lg n)$ and then we'll insert them one by one to the linked list where the sorted order is always maintained. This process will take $O(n)$ and so the entire process can be done in $\Theta(n \log n)$ time.
2. For the latter case, where we have to do insertion to the linked list one element at a time, the only way to ensure sorted order for the linkedlist is to follow insertion sort mechanism. This will take $\Theta(n^2)$ time in the worst case.

Since, the question is ambiguous in saying how the elements are presented, both options A and B should be given marks. Unfortunately official key considered later case only. Option C.

👍 26 votes -- Arjun Suresh (330k points)

3.12.19 Linked Lists: GATE IT 2004 | Question: 13 top

☛ https://gateoverflow.in/3654

✔ In the worst case x could be last or second last node, In that case full traversal of the list is required. Therefore answer is (A).

PS: We can simulate the deletion by moving the $x's$ next node data to x and then delete $x's$ next node. But this is technically not the same as DELETING NODE x as mentioned in the question though effect is the same as long as there are a constant number of elements to be moved from $x's$ next node.

👍 38 votes -- suraj (4.8k points)

3.12.20 Linked Lists: GATE IT 2005 | Question: 54 top

https://gateoverflow.in/3815

✔ It is **(B)** $2, 1, 4, 3, 6, 5, 7$:

As, p and q are swapping each other where q is $p \rightarrow next$ all the time.

24 votes

-- sumit kumar singh dixit (1.6k points)

3.13 Priority Queue (1) top

3.13.1 Priority Queue: GATE CSE 1997 | Question: 4.7 top

https://gateoverflow.in/2248

A priority queue Q is used to implement a stack that stores characters. PUSH (C) is implemented as INSERT (Q, C, K) where K is an appropriate integer key chosen by the implementation. POP is implemented as DELETEMIN (Q). For a sequence of operations, the keys chosen are in

A. non-increasing order
B. non-decreasing order
C. strictly increasing order
D. strictly decreasing order

gate1997 data-structures stack normal priority-queue

Answer

Answers: Priority Queue

3.13.1 Priority Queue: GATE CSE 1997 | Question: 4.7 top

https://gateoverflow.in/2248

✔ Implementing stack using priority queue require first element inserted in stack will be deleted at last, and to implement it using deletemin() operation of queue will require first element inserted in queue must have highest priority.

So the keys must be in strictly decreasing order.

Correct Answer: D

34 votes

-- Suraj Kaushal (273 points)

3.14 Queue (12) top

3.14.1 Queue: GATE CSE 1992 | Question: 09 top

https://gateoverflow.in/588

Suggest a data structure for representing a subset S of integers from 1 to n. Following operations on the set S are to be performed in constant time (independent of cardinality of S).

i.	MEMBER (X) :	Check whether X is in the set S or not
ii.	FIND-ONE (S) :	If S is not empty, return one element of the set S (any arbitrary element will do)
iii.	ADD (X) :	Add integer X to set S
ii.	DELETE (X) :	Delete integer X from S

Give pictorial examples of your data structure. Give routines for these operations in an English like language. You may assume that the data structure has been suitable initialized. Clearly state your assumptions regarding initialization.

gate1992 data-structures normal descriptive queue

Answer

3.14.2 Queue: GATE CSE 1994 | Question: 26 top

https://gateoverflow.in/2522

A queue Q containing n items and an empty stack S are given. It is required to transfer all the items from the queue to the stack, so that the item at the front of queue is on the TOP of the stack, and the order of all other items are preserved. Show how this can be done in $O(n)$ time using only a constant amount of additional storage. Note that the only operations which can be performed on the queue and stack are Delete, Insert, Push and Pop. Do not assume any implementation of the queue or stack.

gate1994 data-structures queue stack normal descriptive

Answer

3.14.3 Queue: GATE CSE 1996 | Question: 1.12 top

https://gateoverflow.in/2716

Consider the following statements:

i. First-in-first out types of computations are efficiently supported by STACKS.
ii. Implementing LISTS on linked lists is more efficient than implementing LISTS on an array for almost all the basic LIST operations.
iii. Implementing QUEUES on a circular array is more efficient than implementing QUEUES on a linear array with two indices.
iv. Last-in-first-out type of computations are efficiently supported by QUEUES.

A. (ii) and (iii) are true
B. (i) and (ii) are true
C. (iii) and (iv) are true
D. (ii) and (iv) are true

gate1996 data-structures easy queue stack linked-lists

Answer

3.14.4 Queue: GATE CSE 2001 | Question: 2.16 top

https://gateoverflow.in/734

What is the minimum number of stacks of size n required to implement a queue of size n?

A. One
B. Two
C. Three
D. Four

gate2001-cse data-structures easy stack queue

Answer

3.14.5 Queue: GATE CSE 2006 | Question: 49 top

https://gateoverflow.in/1826

An implementation of a queue Q, using two stacks $S1$ and $S2$, is given below:

```
void insert (Q, x) {
    push (S1, x);
}
void delete (Q) {
    if (stack-empty(S2)) then
        if (stack-empty(S1)) then {
            print("Q is empty");
            return;
        }
        else while (!(stack-empty(S1))){
            x=pop(S1);
            push(S2,x);
        }
    x=pop(S2);
}
```

let n *insert* and $m(\leq n)$ *delete* operations be performed in an arbitrary order on an empty queue Q. Let x and y be the number of *push* and *pop* operations performed respectively in the process. Which one of the following is true for all m and n?

A. $n+m \leq x < 2n$ and $2m \leq y \leq n+m$
B. $n+m \leq x < 2n$ and $2m \leq y \leq 2n$
C. $2m \leq x < 2n$ and $2m \leq y \leq n+m$
D. $2m \leq x < 2n$ and $2m \leq y \leq 2n$

gate2006-cse data-structures queue stack normal

Answer

3.14.6 Queue: GATE CSE 2012 | Question: 35 top ☛ https://gateoverflow.in/1756

Suppose a circular queue of capacity $(n-1)$ elements is implemented with an array of n elements. Assume that the insertion and deletion operations are carried out using REAR and FRONT as array index variables, respectively. Initially, $REAR = FRONT = 0$. The conditions to detect queue full and queue empty are

A. full: $(REAR+1) \mod n == FRONT$
empty: $REAR == FRONT$

B. full: $(REAR+1) \mod n == FRONT$
empty: $(FRONT+1) \mod n == REAR$

C. full: $REAR == FRONT$
empty: $(REAR+1) \mod n == FRONT$

D. full: $(FRONT+1) \mod n == REAR$
empty: $REAR == FRONT$

gate2012-cse data-structures queue normal

Answer

3.14.7 Queue: GATE CSE 2013 | Question: 44 top ☛ https://gateoverflow.in/61

Consider the following operation along with Enqueue and Dequeue operations on queues, where k is a global parameter.

```
MultiDequeue(Q){
    m = k
    while (Q is not empty) and (m > 0) {
        Dequeue(Q)
        m = m - 1
    }
}
```

What is the worst case time complexity of a sequence of n queue operations on an initially empty queue?

A. $\Theta(n)$
B. $\Theta(n+k)$
C. $\Theta(nk)$
D. $\Theta(n^2)$

gate2013-cse data-structures algorithms normal queue

Answer

3.14.8 Queue: GATE CSE 2016 Set 1 | Question: 10 top ☛ https://gateoverflow.in/39667

A queue is implemented using an array such that ENQUEUE and DEQUEUE operations are performed efficiently. Which one of the following statements is **CORRECT** (n refers to the number of items in the queue) ?

A. Both operations can be performed in $O(1)$ time.
B. At most one operation can be performed in $O(1)$ time but the worst case time for the operation will be $\Omega(n)$.
C. The worst case time complexity for both operations will be $\Omega(n)$.
D. Worst case time complexity for both operations will be $\Omega(\log n)$

gate2016-cse-set1 data-structures queue normal

Answer

3.14.9 Queue: GATE CSE 2016 Set 1 | Question: 41 top ☛ https://gateoverflow.in/39684

Let Q denote a queue containing sixteen numbers and S be an empty stack. $Head(Q)$ returns the element at the head of the queue Q without removing it from Q. Similarly $Top(S)$ returns the element at the top of S without removing it from S. Consider the algorithm given below.

```
while Q is not Empty do
  if S is Empty OR Top(S) ≤ Head (Q) then
```

Shera Q. DID IT ALONE

```
        x:= Dequeue (Q);
        Push (S, x);
    else
        x:= Pop(S);
        Enqueue (Q, x);
    end
  end
```

The maximum possible number of iterations of the **while** loop in the algorithm is _______.

gate2016-cse-set1 data-structures queue difficult numerical-answers

Answer

3.14.10 Queue: GATE CSE 2017 Set 2 | Question: 13 top

☛ https://gateoverflow.in/118253

A circular queue has been implemented using a singly linked list where each node consists of a value and a single pointer pointing to the next node. We maintain exactly two external pointers FRONT and REAR pointing to the front node and the rear node of the queue, respectively. Which of the following statements is/are CORRECT for such a circular queue, so that insertion and deletion operations can be performed in $O(1)$ time?

I. Next pointer of front node points to the rear node.
II. Next pointer of rear node points to the front node.

A. (I) only.
B. (II) only.
C. Both (I) and (II).
D. Neither (I) nor (II).

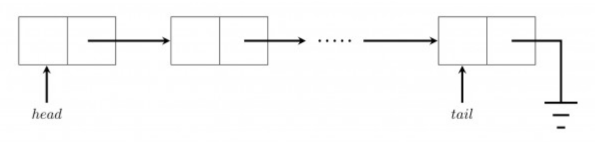
gate2017-cse-set2 data-structures queue

Answer

3.14.11 Queue: GATE CSE 2018 | Question: 3 top

☛ https://gateoverflow.in/204077

A queue is implemented using a non-circular singly linked list. The queue has a head pointer and a tail pointer, as shown in the figure. Let n denote the number of nodes in the queue. Let 'enqueue' be implemented by inserting a new node at the head, and 'dequeue' be implemented by deletion of a node from the tail.



Which one of the following is the time complexity of the most time-efficient implementation of 'enqueue' and 'dequeue, respectively, for this data structure?

A. $\Theta(1), \Theta(1)$
B. $\Theta(1), \Theta(n)$
C. $\Theta(n), \Theta(1)$
D. $\Theta(n), \Theta(n)$

gate2018-cse algorithms data-structures queue normal linked-lists

Answer

3.14.12 Queue: GATE IT 2007 | Question: 30 top

☛ https://gateoverflow.in/3463

Suppose you are given an implementation of a queue of integers. The operations that can be performed on the queue are:

i. $isEmpty(Q)$ — returns true if the queue is empty, false otherwise.
ii. $delete(Q)$ — deletes the element at the front of the queue and returns its value.
iii. $insert(Q, i)$ — inserts the integer i at the rear of the queue.

Consider the following function:

```
void f (queue Q) {
int i ;
```

```
if (!isEmpty(Q)) {
   i = delete(Q);
   f(Q);
   insert(Q, i);
  }
}
```

What operation is performed by the above function f ?

A. Leaves the queue Q unchanged
B. Reverses the order of the elements in the queue Q
C. Deletes the element at the front of the queue Q and inserts it at the rear keeping the other elements in the same order
D. Empties the queue Q

gate2007-it data-structures queue normal

Answer

Answers: Queue

3.14.1 Queue: GATE CSE 1992 | Question: 09 top https://gateoverflow.in/588

A queue with a hashtable.

Intialize hashtable with 0.

When inserting X into the queue update $hashtable[X] = 0$ to $hashtable[X] = 1$.

i. If $hashtable[X] = 1$ then return true.
ii. Return the element at the front or rear of the queue.
iii. Add the element X to the queue at the rear end and update $hashtable[X] = 0$ to $hashtable[X] = 1$.
iv. Delete the element X from the front end of the queue and update $hashtable[X] = 1$ to $hashtable[X] = 0$.

22 votes -- Rajarshi Sarkar (27.8k points)

3.14.2 Queue: GATE CSE 1994 | Question: 26 top https://gateoverflow.in/2522

✔ We can do this be first extracting items one by one from Q, and inserting them to S. After all items are done, S will contain the items in reverse order. Now, pop the elements from S and insert to Q. After this operation, items in Q will be in reverse order from the starting. Now, extract items from Q and push on to stack and we are done.

```
Do
Delete an item from Q
Push the item to S
While (! empty Q);
Do
Pop an item from S
Insert the item to Q
While (! empty S);
Do
Delete an item from Q
Push the item to S
While (! empty Q);
```

48 votes -- Arjun Suresh (330k points)

3.14.3 Queue: GATE CSE 1996 | Question: 1.12 top https://gateoverflow.in/2716

✔ (**A**). (i) and (iv) are false.

http://en.wikipedia.org/wiki/List_(abstract_data_type)#Operations

References

26 votes -- Gate Keeda (15.9k points)

3.14.4 Queue: GATE CSE 2001 | Question: 2.16 top https://gateoverflow.in/734

✔ **A queue can be implemented using two stacks.**

Let queue be represented as " q "
and stacks used to implement q be "stack1" and "stack2".

q can be implemented in two ways:

Method 1 (By making EnQueue operation costly)

This method makes sure that newly entered element is always at the bottom of stack 1, so that deQueue operation just pops from stack1. To put the element at top of stack1, stack2 is used.

$enQueue(q, x)$

1. While stack1 is not empty, push everything from stack1 to stack2.
2. Push x to stack1 (assuming size of stacks is unlimited).
3. Push everything back to stack1.

$dnQueue(q)$

1. If stack1 is empty then error
2. Pop an item from stack1 and return it

Method 2 (By making deQueue operation costly)

In this method, in en-queue operation, the new element is entered at the top of stack1. In de-queue operation, if stack2 is empty then all the elements are moved to stack2 and finally top of stack2 is returned.
$enQueue(q, x)$

1. Push x to stack1 (assuming size of stacks is unlimited).

$deQueue(q)$

1. If both stacks are empty then error.
2. If stack2 is empty
   While stack1 is not empty, push everything from stack1 to stack2.
3. Pop the element from stack2 and return it.

Correct Answer: B

28 votes -- Dipak Majhi (757 points)

3.14.5 Queue: GATE CSE 2006 | Question: 49 top https://gateoverflow.in/1826

✔ Answer is **(A).**

The order in which insert and delete operations are performed matters here.

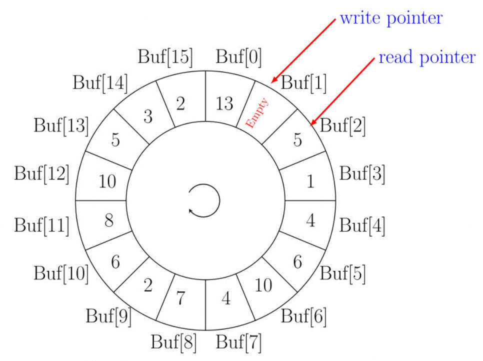
The best case: Insert and delete operations are performed alternatively. In every delete operation, 2 pop and 1 push operations are performed. So, total $m+n$ push $(n$ push for insert() and m push for delete()$)$ operations and $2m$ pop operations are performed.

The worst case: First n elements are inserted and then m elements are deleted. In first delete operation, $n+1$ pop operations and n push operation are performed. Other than first, in all delete operations, 1 pop operation is performed. So, total $m+n$ pop operations and $2n$ push operations are performed $(n$ push for insert() and m push for delete()$)$

52 votes -- Kalpna Bhargav (2.5k points)

3.14.6 Queue: GATE CSE 2012 | Question: 35 top https://gateoverflow.in/1756

✔



$\text{REAR} = \text{Write}$

$\text{FRONT} = \text{Read}$

full: $(\text{REAR} + 1) \mod n == \text{FRONT}$

empty: $\text{REAR} == \text{FRONT}$

Only option $(\mathbf{A})$ matches.

👍 37 votes -- Prashant Singh (47.1k points)

3.14.7 Queue: GATE CSE 2013 | Question: 44 top ☛ https://gateoverflow.in/61

✔ There are three possible operations on queue- Enqueue, Dequeue and MultiDequeue. MultiDequeue is calling Dequeue multiple times based on a global variable k. Since, the queue is initially empty, whatever be the order of these operations, there cannot be more number of Dequeue operations than Enqueue operations. Hence, the total number operations will be n only.

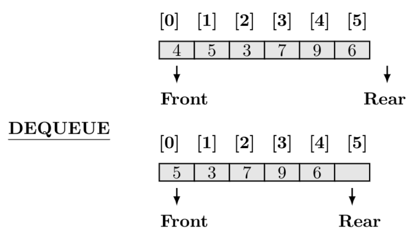
Correct Answer: A

👍 112 votes -- Arjun Suresh (330k points)

3.14.8 Queue: GATE CSE 2016 Set 1 | Question: 10 top ☛ https://gateoverflow.in/39667

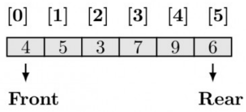
✔ Consider a normal array implementation of a queue. We'll have 2 pointers Front and Rear where Front points to the first element of the queue and Rear points to the next free space. When queue is full Rear points to NULL.

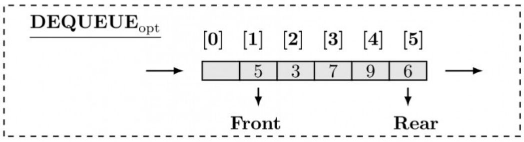
The below figure depicts the Queue full condition and then a DEQUEUE is performed which removes element 4. In the second part of the figure, all elements are shifted by one position or else we won't be able to make use of the free space for the next element to be inserted. So, this means $\Theta(n)$ operations for DEQUEUE where as ENQUEUE can be done in $O(1)$.



But we can in fact do both ENQUEUE and DEQUEUE operations in $O(1)$ and fully utlizie the array space by smartly using the Front and Rear pointers as shown in $\text{DEQUEUE}_{\text{opt}}$. If MAX denote the total size of the array, here,

- for ENQUEUE
  - $\text{Rear} = (\text{Rear} + 1) \mod \text{MAX}$
- for DEQUEUE
  - $\text{Front} = (\text{Front} + 1) \mod \text{MAX}$
- Condition for QUEUE Empty
  - $\text{Front} == \text{NULL}$ (Whenever after a DEQUEUE operation Front becomes equal to Rear it means Queue is now empty and we make $\text{Front} = \text{NULL}$ to mark this condition as $\text{Rear} = \text{Front}$ is the condition we use for checking if QUEUE is full
- Condition for QUEUE Full
  - $\text{Rear} == \text{NULL}$





So, correct answer: A.

1 votes -- Arjun Suresh (330k points)

3.14.9 Queue: GATE CSE 2016 Set 1 | Question: 41 top

https://gateoverflow.in/39684

✔ While loop will run for the maximum number of times when the Queue elements are sorted in descending order.

Let's suppose that initially, Queue elements are $16, 15, 14, 13, \ldots, 2, 1$

Now, 16 will be first pushed into the stack, So, now stack top is 16 and $Head(Q)$ is 15, So 16 will be popped out of the stack (since, "if S is Empty OR $Top(S) \leq Head(Q)$ "returns false, hence else part will be executed) and enqueued into the Queue.

So, after two iterations Queue elements will be $\rightarrow 15, 14, 13, \ldots, 2, 1, 16$ and stack will be empty.

Similarly, each of the elements $15, 14, 13, \ldots, 2$ will be pushed into the stack and immediately popped out of the stack(when popped out of the stack then also enqueue into the queue).So after 30 iterations stack will be empty and Queue contains will be like $\Rightarrow 1, 16, 15, 14, \ldots, 2$.

Now 1 will be Dequeued and pushed into the stack. Once 1 is pushed into the stack, it will never be popped(or we can say never be enqueued into the Queue again) because in order to Pop 1, there should be an element into the Queue which is less than 1 and that element comes at the front of the queue, since there is no element currently present in the Queue which is less than 1, there is no way to pop 1.

So, after 31 iterations Queue is $\Rightarrow 16, 15, 14, \ldots, 2$ and stack contains 1.

Now, the problem boils down to Queue with 15 elements.

Using the similar logic we can say after another 29 iterations (Total $= 31 + 29$)Queue will be like $\Rightarrow 16, 15, 14, \ldots, 3$ and stack contains $1, 2$ (stack top is 2) and then 2 will remain there in the stack forever.

Similar way if we go on then, after $31 + 29 + 27 + 25 + \ldots + 1$ iterations Queue will be empty.

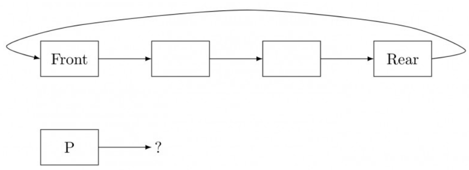
This is in A.P. series with $d = 2$. Sum $= (16 * (1 + 31))/2 = 16 * 32/2 = 256$

176 votes -- Sourav Basu (2.7k points)

3.14.10 Queue: GATE CSE 2017 Set 2 | Question: 13 top

https://gateoverflow.in/118253

✔



Reference: https://gateoverflow.in/1033/gate2004-36

This is how the things look.

We do insertion by cutting in between Rear and Front and we do deletion by forwarding the Front pointer and updating the Rear accordingly.

Correct Answer: B

References

👍 10 votes -- Shamik Banerjee (1.2k points)

3.14.11 Queue: GATE CSE 2018 | Question: 3 top https://gateoverflow.in/204077

✔ New node to be inserted is P.

```
Enqueue(){
    P->Data=Data
    P->Next=Head
    Head=P
}
```

Time Complexity $= O(1)$ Because only pointer manipulation is involved which takes constant time.

Delete Tail

```
Dequeue(){
    temp=head
    While(temp->Next->Next!=NULL)
         temp=temp->next
    temp->next=NULL
    tail=temp
}
```

Time Complexity = Time for Traversing list, free the last node and keep track of Tail pointer $= O(n)$

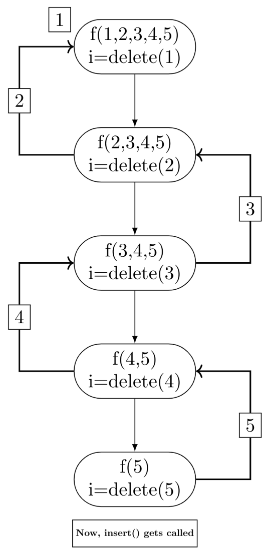
Answer is B

👍 33 votes -- Digvijay (44.9k points)

3.14.12 Queue: GATE IT 2007 | Question: 30 top https://gateoverflow.in/3463

✔

Suppose Q contains 1,2,3,4,5



$insert()$ will inserts the values in reverse order.

Correct Answer: $B-$ Reverses the order of the elements in the queue Q.

👍 36 votes -- srestha (85.2k points)

3.15 Stack (16) top

3.15.1 Stack: GATE CSE 1991 | Question: 03,vii top https://gateoverflow.in/522

The following sequence of operations is performed on a stack:

$PUSH(10), PUSH(20), POP, PUSH(10), PUSH(20), POP, POP, POP, PUSH(20), POP$

The sequence of values popped out is

A. $20, 10, 20, 10, 20$
B. $20, 20, 10, 10, 20$
C. $10, 20, 20, 10, 20$
D. $20, 20, 10, 20, 10$

gate1991 data-structures stack easy multiple-selects

Answer

3.15.2 Stack: GATE CSE 1994 | Question: 1.14 top

https://gateoverflow.in/2457

Which of the following permutations can be obtained in the output (in the same order) using a stack assuming that the input is the sequence $1, 2, 3, 4, 5$ in that order?

A. $3, 4, 5, 1, 2$
B. $3, 4, 5, 2, 1$
C. $1, 5, 2, 3, 4$
D. $5, 4, 3, 1, 2$

gate1994 data-structures stack normal

Answer

3.15.3 Stack: GATE CSE 1995 | Question: 2.21 top

https://gateoverflow.in/2633

The postfix expression for the infix expression $A + B * (C + D)/F + D * E$ is:

A. $AB + CD + *F/D + E*$
B. $ABCD + *F/DE * ++$
C. $A * B + CD/F * DE ++$
D. $A + *BCD/F * DE ++$

gate1995 data-structures stack easy

Answer

3.15.4 Stack: GATE CSE 2000 | Question: 13 top

https://gateoverflow.in/684

Suppose a stack implementation supports, in addition to PUSH and POP, an operation REVERSE, which reverses the order of the elements on the stack.

A. To implement a queue using the above stack implementation, show how to implement ENQUEUE using a single operation and DEQUEUE using a sequence of 3 operations.
B. The following post fix expression, containing single digit operands and arithmetic operators $+$ and $*$, is evaluated using a stack.
   $5\ 2 * 3\ 4 + 5\ 2 * * +$
   Show the contents of the stack
   i. After evaluating $5\ 2 * 3\ 4 +$
   ii. After evaluating $5\ 2 * 3\ 4 + 5\ 2$
   iii. At the end of evaluation

gate2000-cse data-structures stack normal descriptive

Answer

3.15.5 Stack: GATE CSE 2003 | Question: 64 top

https://gateoverflow.in/951

Let **S** be a stack of size $n \geq 1$. Starting with the empty stack, suppose we push the first n natural numbers in sequence, and then perform n pop operations. Assume that Push and Pop operations take X seconds each, and Y seconds elapse between the end of one such stack operation and the start of the next operation. For $m \geq 1$, define the stack-life of m as the time elapsed from the end of $Push(m)$ to the start of the pop operation that removes m from **S**. The average stack-life of an element of this stack is

A. $n(X+Y)$
B. $3Y+2X$
C. $n(X+Y)-X$
D. $Y+2X$

gate2003-cse data-structures stack normal

Answer

3.15.6 Stack: GATE CSE 2004 | Question: 3 top

https://gateoverflow.in/1000

A single array $A[1\dots \text{MAXSIZE}]$ is used to implement two stacks. The two stacks grow from opposite ends of the array. Variables $top1$ and $top2$ $(top1 < top2)$ point to the location of the topmost element in each of the stacks. If the space is to be used efficiently, the condition for "stack full" is

A. $(top1 = \text{MAXSIZE}/2)$ and $(top2 = \text{MAXSIZE}/2+1)$
B. $top1 + top2 = \text{MAXSIZE}$
C. $(top1 = \text{MAXSIZE}/2)$ or $(top2 = \text{MAXSIZE})$
D. $top1 = top2 - 1$

gate2004-cse data-structures stack easy

Answer

3.15.7 Stack: GATE CSE 2004 | Question: 38, ISRO2009-27 top

https://gateoverflow.in/1035

Assume that the operators $+, -, \times$ are left associative and $\hat{}$ is right associative. The order of precedence (from highest to lowest) is $\hat{}, \times, +, -$. The postfix expression corresponding to the infix expression $a+b\times c-d\,\hat{}\,e\,\hat{}\,f$ is

A. $abc\times+def\,\hat{}\,\hat{}\,-$
B. $abc\times+de\,\hat{}\,f\,\hat{}\,-$
C. $ab+c\times d-e\,\hat{}\,f\,\hat{}$
D. $-+a\times bc\,\hat{}\,\hat{}\,def$

gate2004-cse stack isro2009

Answer

3.15.8 Stack: GATE CSE 2004 | Question: 5 top

https://gateoverflow.in/1002

The best data structure to check whether an arithmetic expression has balanced parentheses is a

A. queue
B. stack
C. tree
D. list

gate2004-cse data-structures easy stack

Answer

3.15.9 Stack: GATE CSE 2007 | Question: 38, ISRO2016-27 top

https://gateoverflow.in/1236

The following postfix expression with single digit operands is evaluated using a stack:

$$8\ 2\ 3\ \hat{}\ /\ 2\ 3 * + 5\ 1 * -$$

Note that $\hat{}$ is the exponentiation operator. The top two elements of the stack after the first $*$ is evaluated are

A. $6, 1$
B. $5, 7$
C. $3, 2$
D. $1, 5$

gate2007-cse data-structures stack normal isro2016

Answer

3.15.10 Stack: GATE CSE 2014 Set 2 | Question: 41 top https://gateoverflow.in/2007

Suppose a stack implementation supports an instruction REVERSE, which reverses the order of elements on the stack, in addition to the PUSH and POP instructions. Which one of the following statements is TRUE (*with respect to this modified stack*)?

A. A queue cannot be implemented using this stack.
B. A queue can be implemented where ENQUEUE takes a single instruction and DEQUEUE takes a sequence of two instructions.
C. A queue can be implemented where ENQUEUE takes a sequence of three instructions and DEQUEUE takes a single instruction.
D. A queue can be implemented where both ENQUEUE and DEQUEUE take a single instruction each.

gate2014-cse-set2 data-structures stack easy

Answer

3.15.11 Stack: GATE CSE 2015 Set 2 | Question: 38 top https://gateoverflow.in/8164

Consider the C program below

```
#include <stdio.h>
int *A, stkTop;
int stkFunc (int opcode, int val)
{
    static int size=0, stkTop=0;
    switch (opcode) {
        case -1: size = val; break;
        case 0: if (stkTop < size ) A[stkTop++]=val; break;
        default: if (stkTop) return A[--stkTop];
    }
    return -1;
}
int main()
{
    int B[20]; A=B; stkTop = -1;
    stkFunc (-1, 10);
    stkFunc (0, 5);
    stkFunc (0, 10);
    printf ("%d\n", stkFunc(1, 0)+ stkFunc(1, 0));
}
```

The value printed by the above program is ________.

gate2015-cse-set2 data-structures stack easy numerical-answers

Answer

3.15.12 Stack: GATE CSE 2015 Set 3 | Question: 12 top https://gateoverflow.in/8408

The result evaluating the postfix expression $10\ 5 + 60\ 6 / * 8 -$ is

A. 284
B. 213
C. 142
D. 71

gate2015-cse-set3 data-structures stack normal

Answer

3.15.13 Stack: GATE CSE 2021 Set 1 | Question: 21 top https://gateoverflow.in/357430

Consider the following sequence of operations on an empty stack.

$$\mathsf{push}(54); \mathsf{push}(52); \mathsf{pop}(); \mathsf{push}(55); \mathsf{push}(62); \mathsf{s} = \mathsf{pop}();$$

Consider the following sequence of operations on an empty queue.

$$\text{enqueue}(21); \text{enqueue}(24); \text{dequeue}(); \text{enqueue}(28); \text{enqueue}(32); \text{q} = \text{dequeue}();$$

The value of s+q is __________.

gate2021-cse-set1 data-structures stack numerical-answers

Answer

3.15.14 Stack: GATE IT 2004 | Question: 52 top

https://gateoverflow.in/3695

A program attempts to generate as many permutations as possible of the string, ' $abcd$' by pushing the characters a, b, c, d in the same order onto a stack, but it may pop off the top character at any time. Which one of the following strings CANNOT be generated using this program?

A. $abcd$
B. $dcba$
C. $cbad$
D. $cabd$

gate2004-it data-structures normal stack

Answer

3.15.15 Stack: GATE IT 2005 | Question: 13 top

https://gateoverflow.in/3758

A function f defined on stacks of integers satisfies the following properties. $f(\emptyset) = 0$ and $f(push(S, i)) = max(f(S), 0) + i$ for all stacks S and integers i.

If a stack S contains the integers $2, -3, 2, -1, 2$ in order from bottom to top, what is $f(S)$?

A. 6
B. 4
C. 3
D. 2

gate2005-it data-structures stack normal

Answer

3.15.16 Stack: GATE IT 2007 | Question: 32 top

https://gateoverflow.in/3465

Consider the following C program:

```
#include <stdio.h>
         #define EOF -1
         void push (int); /* push the argument on the stack */
         int pop  (void); /* pop the top of the stack */
         void flagError ();
         int main ()
        {         int c, m, n, r;
                   while ((c = getchar ()) != EOF)
                  { if  (isdigit (c) )
                             push (c);
                   else if ((c == '+') || (c == '*'))
                  {          m = pop ();
                              n = pop ();
                              r = (c == '+') ? n + m : n*m;
                              push (r);
                    }
                    else if (c != ' ')
                             flagError ();
             }
              printf("% c", pop ());
}
```

What is the output of the program for the following input?
$5\ 2\ *\ 3\ 3\ 2\ +*+$

A. 15
B. 25
C. 30
D. 150

gate2007-it stack normal

Answer

Answers: Stack

3.15.1 Stack: GATE CSE 1991 | Question: 03,vii top

https://gateoverflow.in/522

✔ Let us try something different when you read the word pop then delete the last pushed element and print it. Now, delete the push word which we have already executed. Now, go on from left to right and do the same.

So, output will be $20, 20, 10, 10, 20.$

Correct Answer: B.

31 votes -- Bhagirathi Nayak (11.7k points)

3.15.2 Stack: GATE CSE 1994 | Question: 1.14 top

https://gateoverflow.in/2457

✔ Push 1 push 2 push 3 pop 3 push 4 pop 4 push 5 pop 5 pop 2 pop 1 then o/p is $3, 4, 5, 2, 1$

So, option is **B.**

40 votes -- Sankaranarayanan P.N (8.5k points)

3.15.3 Stack: GATE CSE 1995 | Question: 2.21 top

https://gateoverflow.in/2633

✔

(A + B * (C + D) / F + D * E)

Thus before considering + which has the least priority, we get $A + (BCD + *F/) + (DE*)$

Now if we assume left associativity for $+$ (default), we get $ABCD + *F/ + DE * +$ but this is not among the options.

So, considering right associativity for $+$ we get $ABCD + *F/DE * ++$

Correct Answer: B

48 votes -- Amar Vashishth (25.2k points)

3.15.4 Stack: GATE CSE 2000 | Question: 13 top

https://gateoverflow.in/684

✔

a. For enqueue push operation is sufficient
   For dequeue operation do the following
   -reverse, pop, reverse

b. Contents of stack from top to bottom:

   i) 7 10
   ii) 2 5 7 10
   ii) 80

31 votes -- Pooja Palod (24.1k points)

3.15.5 Stack: GATE CSE 2003 | Question: 64 top

https://gateoverflow.in/951

✔ Let us represent stack-life of i^{th} element as S(i). The i^{th} element will be in stack till $(n - i)$ elements are pushed and popped. Plus one more Y for the time interval between the push of i_{th} element and the $i + 1^{th}$ element. So,

$S(i) = Y + 2.(n - i)(Y + X) = Y + 2.(n - i)Z = Y + 2nZ - 2iZ$

where $Z = Y + X$

average stack-life will, $A = \Sigma \frac{S(i)}{n}$

$nA = nY + 2.n.n.Z - 2.Z.\Sigma i$

$nA = nY + 2.n.n.Z - 2.Z\frac{(n(n+1))}{2}$

$nA = nY + 2.n.n.Z - Z(n.n) - n.Z$

$A = Y + 2.n.Z - (n+1).Z$

$A = Y + (n-1).Z = Y + (n-1)(X+Y) = n(X+Y) - X$

Correct Answer: C

63 votes -- Vikrant Singh (11.2k points)

3.15.6 Stack: GATE CSE 2004 | Question: 3 top https://gateoverflow.in/1000

✔ Answer is (**D**).

Since the stacks are growing from opposite ends, initially $top1 = 1$ and $top2 = MAXSIZE$. Now, to make the space usage most efficient we should allow one stack to use the maximum possible space as long as other stack doesn't need it. So, either of the stack can grow as long as there is space on the array and hence the condition must be $top1 = top2 - 1$.

52 votes -- Aditi Dan (4k points)

3.15.7 Stack: GATE CSE 2004 | Question: 38, ISRO2009-27 top https://gateoverflow.in/1035

✔ Answer is (**A**).

Here is the procedure first :
Scan Infix Expression from left to right whenever you see operand just print it.
But, in case of operator
if(stack is empty) then push it.
if(precedence(tos) $<$ precedence(current operator)) push it. where, tos means top of stack.
else if (precedence(tos) $>$ precedence(current operator)) pop and print.
else if (precedence(tos) $==$ precedence(current operator)) then check for associativity.In case Operators are Left to right then pop and print it otherwise push the current operator (Right to Left Associativity)
And once you have scanned infix expression completely. Make sure pop all the element and print it in same order.

Here, the infix expression is $a + b \times c - d\hat{}e\hat{}f$
a : print it
$+$: push into the Operator Stack
b : print it
$*$: its having higher precedence than $+$ then push into Operator Stack
c : print it
$'-'$: $'-'$ is having less precedence than $'*'$ so pop from operator stack and print $'*'$.after this stack will be having $'+'$ on top.which is having same precedence as $'-'$ but both are left to right associative then just pop $+$ and print it. Now stack is empty. Push $'-'$ to it.
d : print it
'^' : top of the stack is having $'-'$.'^' has higher precedence than $'-'$.so simply push '^' into stack
e : print it.
'^' : Now top of the stack is '^' and has same precedence so associativity will come to picture. Since '^' is right associative as given in question. So '^' will be pushed.
f : print it.

Now, we have scanned entire infix expression.Now pop the stack until it becomes empty.This way you will get $abc * +def\hat{}\,\hat{}-$.

59 votes -- IgnitorSandeep (345 points)

3.15.8 Stack: GATE CSE 2004 | Question: 5 top https://gateoverflow.in/1002

✔ STACK scan the expression from left to right whenever a left paranthesis is encountered just PUSH it into stack and whenever a right paranthesis is encountered just POP it from stack. If at the end of expression we are left with an empty stack then it is a correctly parenthesized expression.

30 votes -- Bhagirathi Nayak (11.7k points)

3.15.9 Stack: GATE CSE 2007 | Question: 38, ISRO2016-27 top https://gateoverflow.in/1236

✔ push 8 so stack is 8

push 2 so stack is 8 2

push 8 2 3

^ pop 3 and 2 perform opn 2^3 and push to stack. stack is 8 8

/ pop 8 and 8 perform $8/8$ and push result to stack . stack is 1

push 2 stack is 1 2

push 3 stack is 1 2 3

$*$ pop 3 and 2 perform by $2*3$ and push . stack is 1 6

Hence, answer is **A.**

36 votes -- Sankaranarayanan P.N (8.5k points)

3.15.10 Stack: GATE CSE 2014 Set 2 | Question: 41 top https://gateoverflow.in/2007

✔ Correct Option: **C**

While ENQUEUE we REVERSE the stack, PUSH the element and then again REVERSE the stack. For DEQUE we simply POP the element.

Option **B** can be used to get the first element from the stack by doing a POP after REVERSE for DEQUEUE and PUSH for ENQUEUE. But we have to restore the stack using REVERSE (otherwise next POP won't work) which means DEQUEUE actually needs 3 instructions and not 2.

64 votes -- Arjun Suresh (330k points)

3.15.11 Stack: GATE CSE 2015 Set 2 | Question: 38 top https://gateoverflow.in/8164

✔ Answer: 15

The code is pushing 5 and 10 on stack and then popping the top two elements and printing their sum.

http://ideone.com/kIUdQT

References

32 votes -- Rajarshi Sarkar (27.8k points)

Initially stack is empty = -1

stkFunc (-1, 10); this function

case -1: size = val; break; and static size= 10 // size memory declare one time only// and control comes out of switch b/c of break

stkFunc (0, 5); this function run

case 0: if (stkTop < size) A[stkTop++]=val; break; here stktop is static value so memory declare at compile time only now check if condition 0< 10 true then A[stktop++== A[0+1]=val= 5 i.e. push 5 into stack break comes so control comes outside

stkFunc (0, 10); this comes

case 0: if (stkTop < size) A[stkTop++]=val; break; same as above make A[stkTop++]= 10 i,e. push 10 into stack and break comes so control comes outside

printf ("%d\n", stkFunc(1, 0)+ stkFunc(1, 0));

```
this function
stkFunc(1, 0) this will run
default: if (stkTop) return A[--stkTop] return top of stack which is 10
stkFunc(1, 0) this will run
default: if (stkTop) return A[--stkTop] return top of stack which is 5
printf ("%d\n", stkFunc(1, 0)+ stkFunc(1, 0));= 5+10=15 15 will be printed
```

39 votes -- Prashant Singh (47.1k points)

3.15.12 Stack: GATE CSE 2015 Set 3 | Question: 12 top https://gateoverflow.in/8408

✔ We have to keep symbol into stack and when we get two operands followed by operator. We will apply operator on last two operands.

symbol	stack
10	10 (keep in stack)
5	10 5 (keep in stack)
$+$	10 5 $+$ (apply operator on last 2 operands $\implies$ $10+5=15$)
60	15 60 (keep in stack)
6	15 60 6 (keep in stack)
$/$	15 60 6 $/$ (apply operator on last 2 operands $\implies$ $60/6=10$)
$*$	15 10 $*$ (apply operator on last 2 operands $\implies$ $10*15=150$)
8	150 8 (keep in stack)
$-$	150 8 $-$ (apply operator on last 2 operands $\implies$ $50-8=142$)

So, answer is 142.

43 votes -- Praveen Saini (41.9k points)

3.15.13 Stack: GATE CSE 2021 Set 1 | Question: 21 top https://gateoverflow.in/357430

✔ **Stack:**

- Push 54, push 52, pop (remove top element of stack)
- Now stack 54 (remove 52), push 55, push 62, pop
- Now top element is 62 (remove 62 as S)
- $S=62$

Queue

- Enqueue 21, Enqueue 24 dequeue (remove first element of queue)
- Now queue 24 (remove 21), Enqueue 28, Enqueue 32, dequeue
- Starting element is 24 (remove 24 as Q)
- $Q=24$

$S+Q=62+24=86$

5 votes -- Ankur tiwari (557 points)

3.15.14 Stack: GATE IT 2004 | Question: 52 top https://gateoverflow.in/3695

✔

A. push a & pop a, push b & pop b, push c & pop c, and finally push d and pop d sequence of popped elements will come to $abcd$
B. first push $abcd$, and after that pop one by one, sequence of popped elements will come to $dcba$
C. push abc, and after that pop one by one sequence of popped elements will come to cba, now push d and pop d, final sequence comes to $cbad$
D. this sequence is not possible because 'a' can not be popped before 'b' any how

28 votes -- Manu Thakur (34.1k points)

3.15.15 Stack: GATE IT 2005 | Question: 13 top https://gateoverflow.in/3758

- i : Element to be pushed
- Initial State $f(\emptyset) = 0$. For Empty Stack $f(S)$ is 0
- Then we push each element (i)one by one and calculate $f(s)$ for each insertion as given

$$f_{new}(S) = \max(f_{previous}(S), 0) + i$$

is the function to compute $f(S)$ for each insertions

1. INSERT 2 on to Stack
   $f_{previous}(S) = 0$ [Stack was empty]
   $i = 2$ (inserting element is i)
   $f_{new}(S) = \max(f_{previous}(S), 0) + i$
   $f_{new}(S) = \max(0, 0) + 2 = 2$
2. INSERT -3 on to Stack
   $f_{previous}(S) = 2$ [Stack was empty]
   $i = -3$ (inserting element is i)
   $f_{new}(S) = \max(f_{previous}(S), 0) + i$
   $f_{new}(S) = \max(2, 0) + -3 = -1$

Similarly,

- i: The element to be pushed
- S: Stack
- Initially $f(S) = 0$.

$f(S)$	$\max(f(S), 0)$	i	$f_{new}(S) = \max(f(S), 0) + i$
0	0	2	2
2	2	-3	-1
-1	0	2	2
2	2	-1	1
1	1	2	$\boxed{3}$

Thus, the answer is **Option C.**

The value of $f(S)$ after inserting all elements into stack is **3.**

88 votes -- Shridhar (311 points)

3.15.16 Stack: GATE IT 2007 | Question: 32 top https://gateoverflow.in/3465

Correct Option: **B**

25
let first part
5 ----push
2------push
push------$5 * 2 = 10$. (pops 5 and 2)

push 3
push 3
push 2
push $3 + 2 = 5$ (pops 2 and 3)
push $5 * 3 = 15$ (pops (5 and 3)
push $15 + 10 = 25$ (pops (15 and 10)

26 votes -- Arpit Dhuriya (2.9k points)

3.16 Trees (14) top

3.16.1 Trees: GATE CSE 1990 | Question: 13a top https://gateoverflow.in/86224

Consider the height-balanced tree T_t with values stored at only the leaf nodes, shown in Fig .4.

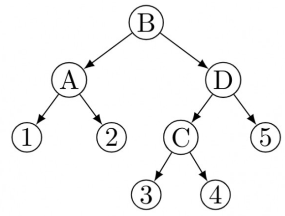


Fig.4

(i) Show how to merge to the tree, T_1 elements from tree T_2 shown in Fig.5 using node D of tree T_1.

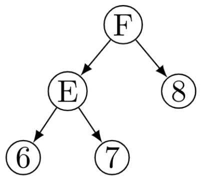


Fig.5

(ii) What is the time complexity of a merge operation of balanced trees T_1 and T_2 where T_1 and T_2 are of height h_1 and h_2 respectively, assuming that rotation schemes are given. Give reasons.

gate1990 data-structures trees descriptive

Answer

3.16.2 Trees: GATE CSE 1992 | Question: 02,vii top https://gateoverflow.in/562

A $2-3$ tree is such that

a. All internal nodes have either 2 or 3 children
b. All paths from root to the leaves have the same length

The number of internal nodes of a $2-3$ tree having 9 leaves could be

A. 4
B. 5
C. 6
D. 7

gate1992 trees data-structures normal multiple-selects

Answer

3.16.3 Trees: GATE CSE 1994 | Question: 5 top https://gateoverflow.in/2501

A $3-$ary tree is a tree in which every internal node has exactly three children. Use induction to prove that the number of leaves in a $3-$ary tree with n internal nodes is $2(n+1)$.

gate1994 data-structures trees proof descriptive

Answer

3.16.4 Trees: GATE CSE 1998 | Question: 1.24 top https://gateoverflow.in/1661

Which of the following statements is false?

A. A tree with a n nodes has $(n-1)$ edges
B. A labeled rooted binary tree can be uniquely constructed given its postorder and preorder traversal results.
C. A complete binary tree with n internal nodes has $(n+1)$ leaves.
D. The maximum number of nodes in a binary tree of height h is $2^{h+1}-1$

gate1998 data-structures trees multiple-selects normal

Answer

3.16.5 Trees: GATE CSE 1998 | Question: 2.11 top https://gateoverflow.in/1683

A complete n-ary tree is one in which every node has 0 or n sons. If x is the number of internal nodes of a complete n-ary tree, the number of leaves in it is given by

A. $x(n-1)+1$
B. $xn-1$
C. $xn+1$
D. $x(n+1)$

gate1998 data-structures trees normal

Answer

3.16.6 Trees: GATE CSE 1998 | Question: 21 top https://gateoverflow.in/1735

A. Derive a recurrence relation for the size of the smallest AVL tree with height h.
B. What is the size of the smallest AVL tree with height 8?

gate1998 data-structures trees descriptive numerical-answers

Answer

3.16.7 Trees: GATE CSE 2002 | Question: 2.9 top https://gateoverflow.in/839

The number of leaf nodes in a rooted tree of n nodes, with each node having 0 or 3 children is:

A. $\frac{n}{2}$
B. $\frac{(n-1)}{3}$
C. $\frac{(n-1)}{2}$
D. $\frac{(2n+1)}{3}$

gate2002-cse data-structures trees normal

Answer

3.16.8 Trees: GATE CSE 2004 | Question: 6 top https://gateoverflow.in/1003

Level order traversal of a rooted tree can be done by starting from the root and performing

A. preorder traversal
B. in-order traversal
C. depth first search
D. breadth first search

gate2004-cse data-structures trees easy

Answer

3.16.9 Trees: GATE CSE 2005 | Question: 36 top https://gateoverflow.in/1372

In a complete k-ary tree, every internal node has exactly k children. The number of leaves in such a tree with n internal node is:

A. nk
B. $(n-1)k+1$
C. $n(k-1)+1$
D. $n(k-1)$

gate2005-cse data-structures trees normal

Answer

3.16.10 Trees: GATE CSE 2007 | Question: 43 top

https://gateoverflow.in/1241

A complete $n-ary$ tree is a tree in which each node has n children or no children. Let I be the number of internal nodes and L be the number of leaves in a complete $n-ary$ tree. If $L = 41$ and $I = 10$, what is the value of n?

A. 3
B. 4
C. 5
D. 6

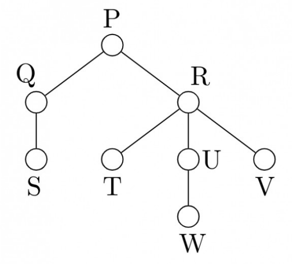
gate2007-cse data-structures trees normal

Answer

3.16.11 Trees: GATE CSE 2014 Set 3 | Question: 12 top

https://gateoverflow.in/2046

Consider the following rooted tree with the vertex labeled P as the root:



The order in which the nodes are visited during an in-order traversal of the tree is

A. $SQPTRWUV$
B. $SQPTUWRV$
C. $SQPTWUVR$
D. $SQPTRUWV$

gate2014-cse-set3 data-structures trees easy

Answer

3.16.12 Trees: GATE CSE 2014 Set 3 | Question: 41 top

https://gateoverflow.in/2075

Consider the pseudocode given below. The function $DoSomething()$ takes as argument a pointer to the root of an arbitrary tree represented by the $leftMostChild - rightSibling$ representation. Each node of the tree is of type $treeNode$.

```
typedef struct treeNode* treeptr;

struct treeNode
{
    treeptr leftMostChild, rightSibling;
};

int DoSomething (treeptr tree)
{
    int value=0;
    if (tree != NULL) {
        if (tree->leftMostChild == NULL)
            value = 1;
        else
        value = DoSomething(tree->leftMostChild);
        value = value + DoSomething(tree->rightSibling);
    }
    return(value);
}
```

When the pointer to the root of a tree is passed as the argument to $DoSomething$, the value returned by the function corresponds to the

A. number of internal nodes in the tree.
B. height of the tree.
C. number of nodes without a right sibling in the tree.
D. number of leaf nodes in the tree

gate2014-cse-set3 data-structures trees normal

Answer

3.16.13 Trees: GATE CSE 2017 Set 1 | Question: 20 top
https://gateoverflow.in/118300

Let T be a tree with 10 vertices. The sum of the degrees of all the vertices in T is ________

gate2017-cse-set1 data-structures trees numerical-answers

Answer

3.16.14 Trees: GATE CSE 2021 Set 1 | Question: 41 top
https://gateoverflow.in/357410

A *n articulation point* in a connected graph is a vertex such that removing the vertex and its incident edges disconnects the graph into two or more connected components.

Let T be a DFS tree obtained by doing DFS in a connected undirected graph G.

Which of the following options is/are correct?

A. Root of T can never be an articulation point in G.
B. Root of T is an articulation point in G if and only if it has 2 or more children.
C. A leaf of T can be an articulation point in G.
D. If u is an articulation point in G such that x is an ancestor of u in T and y is a descendent of u in T, then all paths from x to y in G must pass through u.

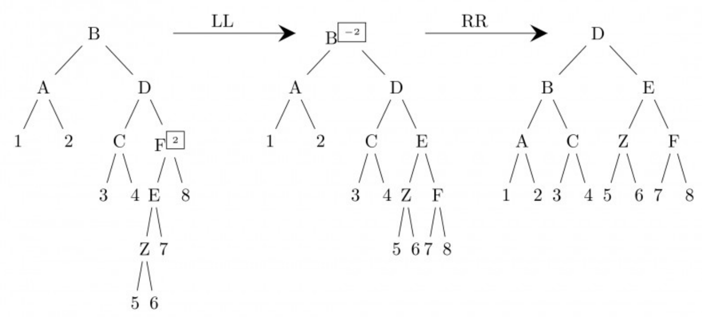
gate2021-cse-set1 multiple-selects data-structures trees

Answer

Answers: Trees

3.16.1 Trees: GATE CSE 1990 | Question: 13a top
https://gateoverflow.in/86224

✔ Superscripts near the nodes indicate their Balance factor. LL and RR rotations are same as those done on AVL trees.



9 votes -- Charan (1.1k points)

3.16.2 Trees: GATE CSE 1992 | Question: 02,vii top
https://gateoverflow.in/562

✔ Correct Options: **A;D**

$4 \rightarrow$ When each leaf has 3 childs. So $9/3 = 3$ Internal nodes, Then one internal node those internal nodes.

$7 \rightarrow$ When each leaf has 2 childs & one leaf out of 4 get 3 childs. Ex $\rightarrow 8/4 = 2$ child per internal node. Then one of that internal node get extra third child. Then 2 internal nodes to connect these 4. Then 1 internal node to connect this 2. So $4 + 2 + 1 = 7$.

No other way is possible.

👍 26 votes -- Akash Kanase (36k points)

3.16.3 Trees: GATE CSE 1994 | Question: 5 top

☛ https://gateoverflow.in/2501

✔ Number of nodes at level $i = 3^i$

Let height of the tree be h

So total no of internal nodes, $n = 3^0 + 3^1 + 3^2 + \dots + 3^{h-1} = \frac{3^h - 1}{2}$

$2n = 3^h - 1$

Number of leaf nodes $= 3^h = 2n + 1 = 2(n-1) + 3$

Let us prove by induction

Base case

$n = 1$ (one internal node i.e., root node)

No of leaves $= 2(1-1) + 3 = 3$

Let it be true for n internal nodes

Now we prove for m nodes where $m = n + 1$

We have $L(m) = 2(n + 1 - 1) + 3$

Also $L(m) = L(n) + 3 - 1 = 2(n-1) + 3 + 3 - 1 = 2n + 3$

So if $L(n)$ is true then $L(n+1)$ is also true

Hence proved by induction.

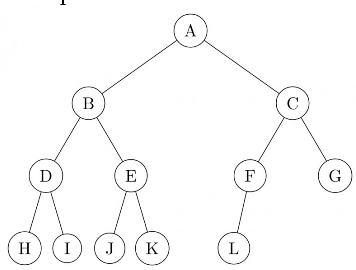
👍 13 votes -- Pooja Palod (24.1k points)

3.16.4 Trees: GATE CSE 1998 | Question: 1.24 top

☛ https://gateoverflow.in/1661

✔

- Tree with n nodes must have $n - 1$ edges.
- A labeled rooted binary tree can be uniquely constructed given its postorder and preorder traversal results. (inorder must be needed with either preorder or postorder for that)
- A complete binary tree with n nodes can have n leaves also
- Example:



Since: **A complete binary tree** is a binary tree in which every level, except possibly the last, is completely filled, and all nodes are as far left as possible. So **false**

- The maximum number of nodes in a binary tree of height h is

$$1 + 2 + 4 + \dots + 2^h = 2^{h+1} - 1 \text{ So true}$$

Answer is **b** and **c** both.

Since, 2 answers are there I would choose b, because in some places by "complete" binary tree they mean "full binary tree" which has all the levels completely filled.

👍 27 votes -- Prashant Singh (47.1k points)

3.16.5 Trees: GATE CSE 1998 | Question: 2.11 top

https://gateoverflow.in/1683

✔ Correct Option: A

$x(n-1)+1$

Originally when we have root , there is only 1 node, which is leaf. (There is no internal node.) **From this base case "+1" part of formula comes.**

When we n children to root, we make root internal. So then Total Leaves $= \; = 1(n-1)+1 = n$.

In complete n ary tree every time you add n children to node, you add $n-1$ leaves & make that node to which you are inserting childen internal.($+n$ for leaves, -1 for node which you are attaching). So if you had originally few leaves, you add $n-1$ "New" leaves to them. This is how $x(n-1)+1$ makes sense.

👍 33 votes -- Akash Kanase (36k points)

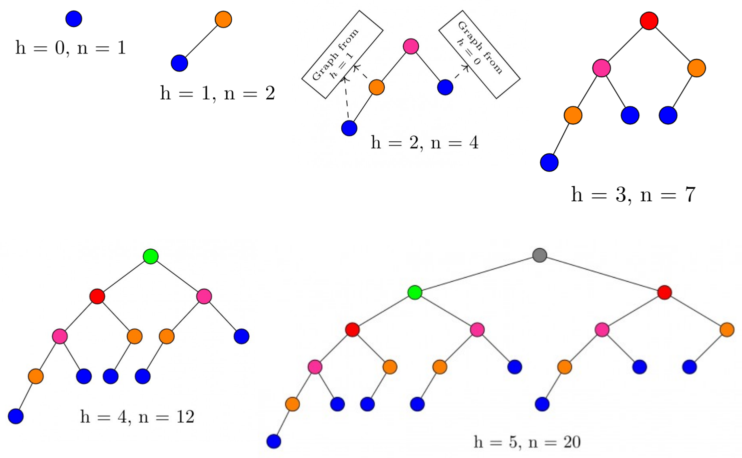
3.16.6 Trees: GATE CSE 1998 | Question: 21 top

https://gateoverflow.in/1735

✔

a. Consider a function $N(h)$ which represents the smallest number of nodes n for an AVL tree with height h and satisfies $n = N(h)$.
For $h = 0$ we have, number of nodes $= 1$. So $N(0) = 1$
For $h = 1$, we have, number of nodes $= 2$. We could take 3, but we require the smallest graph (a graph with smallest number of nodes) so we take 2. It means that to create a tree with height \$14 we need **at least** 24 nodes.
So $N(1) = 2$
Now, for $h = 2$, we need to create a node with a child subtree of height 1. This may be the right or left subtree. But since this is an AVL tree, to balance a child subtree of height let's say Hs, we need the other child to have a height of $Hs-1$, Hs or $Hs+1$. But we take $Hs-1$ for minimal case. In simple words, a node with height 5 must have a child with height $4(Hs)$ and another child with height $3(Hs-1)$. So $N(2)$ can be obtained as:
$N(2) = N(1) + (0) + 1$ (1 is added to count the parent node, $N(1)$ or $N(Hs)$ and $N(0)$ or $N(Hs-1)$ represent two subtrees.)
Similarly:
$N(3) = N(2) + N(1) + 1$
and generalizing:
$\mathbf{N(h) = N(h-1) + N(h-2) + 1}$

b. This recursion can be graphically seen as below:



Using the above recursion, we need to find $N(8)$

$N(0) = 1$
$N(1) = 2$
$N(2) = N(1) + N(0) + 1 = 1 + 2 + 1 = 4$
$N(3) = N(2) + N(1) + 1 = 2 + 4 + 1 = 7$
$N(4) = N(3) + N(2) + 1 = 4 + 7 + 1 = 12$
$N(5) = N(4) + N(3) + 1 = 7 + 12 + 1 = 20$
$N(6) = N(5) + N(4) + 1 = 12 + 20 + 1 = 33$
$N(7) = N(6) + N(5) + 1 = 20 + 33 + 1 = 54$
$N(8) = N(7) + N(6) + 1 = 33 + 54 + 1 = 88$

So, answer for (**b**) is 88.

30 votes -- Ashis Kumar Sahoo (699 points)

3.16.7 Trees: GATE CSE 2002 | Question: 2.9 top

https://gateoverflow.in/839

✔ $L =$ leaf nodes

$I =$ internal nodes

$n =$ total nodes $= L + I$

In a tree no. of edges $= n - 1$

All edges are produced by only internal nodes so,

$k \times I = n - 1 \quad \rightarrow (1)$ (for $k - ary$ tree, in this question $k = 3$)

$L + I = n \quad \rightarrow (2)$

Here, given options are in terms of "n". So, eliminating I from (1) and (2),

$L = ((k - 1)n + 1)/k$

you get $L = (2n + 1)/3$

Answer is **D.**

50 votes -- Vikrant Singh (11.2k points)

3.16.8 Trees: GATE CSE 2004 | Question: 6 top

https://gateoverflow.in/1003

✔ Answer is **option D.**

Breadth first seach.

20 votes -- anshu (2.7k points)

3.16.9 Trees: GATE CSE 2005 | Question: 36 top

https://gateoverflow.in/1372

✔ Correct Option: **C**

Originally when we have root , there is only 1 node, which is leaf.(There is no internal node.) From that " $+1$" part of formula comes from this base case.

When we k children to nodes, we make root internal. So then Total Leaves $= n(k - 1) + 1 = (k - 1) + 1 = k$

In k complete k ary tree every time you add k children , you add $k - 1$ leaves.($+k$ for leaves, -1 for node which you are attaching)

16 votes -- Akash Kanase (36k points)

3.16.10 Trees: GATE CSE 2007 | Question: 43 top

https://gateoverflow.in/1241

✔ If you do little bit experiments on no of leaves, Internal nodes, you will realize that they have equation like following :-

No of leaves $(L) = (n - 1)*$ Internal Nodes $(I) + 1$

Here we need to find n.

Putting values

$$41 = (n - 1) * 10 + 1$$
$$\implies (n - 1) * 10 = 40$$
$$\implies n - 1 = 4$$
$$\implies n = 5$$

So, answer is **C.**

29 votes -- Akash Kanase (36k points)

Sum of degrees in tree $= L + I \times (n + 1) - 1 = 10n + 50$(Each leaf node has degree 1 and all internal nodes have degree $k + 1$, except root which has degree k)

So, number of edges $= 5n + 25$(Number of edges in a graph (hence applicable for tree also) is half the sum of degrees as each edge contribute 2 to the sum of degrees)

In a tree with n nodes we have $n - 1$ edges, so with $41 + 10 = 51$ nodes, there must be 50 edges.

So, $5n + 25 = 50$

$\implies 5n = 25$

$\implies n = 5$

36 votes -- Arjun Suresh (330k points)

3.16.11 Trees: GATE CSE 2014 Set 3 | Question: 12 top https://gateoverflow.in/2046

✔ Correct Option: **A**

The inorder traversal order of a ternary tree is left $\rightarrow$ root $\rightarrow$ middle $\rightarrow$ right.

46 votes -- Gate Keeda (15.9k points)

3.16.12 Trees: GATE CSE 2014 Set 3 | Question: 41 top https://gateoverflow.in/2075

✔ Here, the condition for count value $= 1$ is

if $(tree \rightarrow leftMostchild == Null)$

- so, if there is no left-most child of the tree (or the sub-tree or the current node called in recursion)
- Which means there is no child to that particular node (since if there is no left-most child, there is no child at all as per the tree representation given).
- $\therefore$ the node under consideration is a leaf node.
- The function recursively counts, and adds to value, whenever a leaf node is encountered.

So, The function returns the number of leaf nodes in the tree. Answer is D

32 votes -- Kalpish Singhal (1.6k points)

3.16.13 Trees: GATE CSE 2017 Set 1 | Question: 20 top https://gateoverflow.in/118300

✔ Tree with n vertices which means $n - 1$ edges.

$n = 10 \therefore \ edges = n - 1 = 9$.

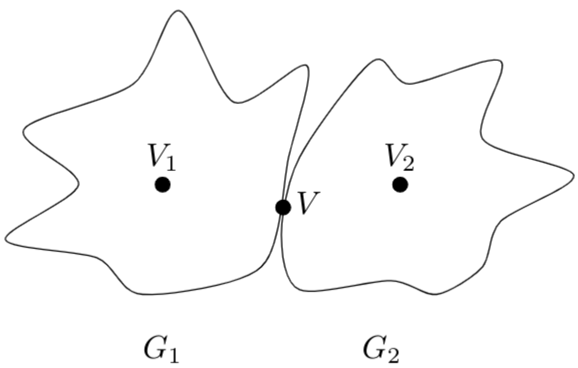
$\therefore$ Sum of degree of all vertices $\leq 2E \leq 2 * 9 \leq 18$

Answer is 18

41 votes -- Kantikumar (3.4k points)

3.16.14 Trees: GATE CSE 2021 Set 1 | Question: 41 top https://gateoverflow.in/357410

✔



As per the option B it says root of T can be an articulation point in G if and only if it has 2 or more children, now here for simplicity I have taken simple case where root will have 2 children and I will tell you for the generalized case later. Also as its double implication lets see it as separately

Case 1: If root is articulation point then root will have 2 or more children

If a vertex is articulation point then its removal disconnects the graph into 2 or more components means there must exist at least 2 vertices for which each and every path in between them will pass through articulation point and removal of articulation point will destroy all the paths in between them and thereby disconnecting the graph into components, otherwise that vertex can not be an articulation point because even if we remove that vertex still every pair of vertices has some other path and hence graph wont get disconnected.

Now when we start DFS Traversal from vertex V (articulation point) we may visit either the vertex from G_1 or G_2 first, lets say we visited vertex of G_2 first now during the DFS traversal which becomes child of V now we can never visit any vertex in G_1 unless and until we do not use vertex V because every path will go through vertex V only, and by the nature of DFS no vertex during the traversal of sub-graph G_2 can visit the vertex V again as it will be having Gray color/ not yet finished (refer DFS algorithm from CLRS for better understanding) and because of this property all the vertices in G_2 will be exhausted and we will be back to the vertex V but vertex V still has path to the unvisited vertices of sub-graph G_1 and hence the first vertex which will be visited in G_1 will become the new child of V thereby making 2 children for the root vertex V which is the articulation point.

Case 2: If root vertex has 2 or more children then it is articulation point

Lets say in an undirected graph if root has 2 children then it is true that there is no path between the vertices in left sub-tree and right sub-tree of vertex V (w.r.t DFS traversal tree) because if there had been any path between the left and right sub-tree the in that case if we start with right child then before reaching to the root all the vertices in left sub-tree would have been visited and root had only single child but it is contradiction as root has 2 children and hence there can be no path between the left and right sub-tree of vertex V, thereby making it the ONLY vertex through which vertices in left and right sub-tree are connected

Hence above two cases proves the option B is correct and A is incorrect

For the generalized case visualize star graph with many vertices where center is articulation point, now you got the intuition and apply this on any graph!

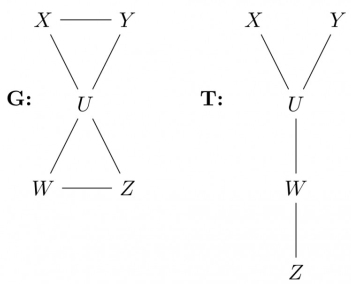
Lets understand option C.

Option C says that leaf of tree T can be an articulation point, its FALSE because if some vertex is leaf of tree T then all the vertices to which it connects are already been visited which indicates that even without using this leaf vertex there exists path between all of its neighbors and hence it can not be an articulation point.

Hence option C is incorrect

Now option D

Option D talks about ancestors and decedents X and Y and says that if X is ancestor of U in T and Y is decedent then all the paths between X and Y must pass through U but we have counter for this as shown below.



Hence option D is incorrect

16 votes -- Jaydeep Vasudev Pawar (295 points)

Answer Keys

3.1.1	C	3.2.1	N/A	3.2.2	C	3.2.3	N/A	3.2.4	N/A
3.2.5	A	3.2.6	A	3.2.7	N/A	3.2.8	A	3.2.9	B
3.2.10	C	3.2.11	B	3.2.12	5	3.2.13	C	3.3.1	B
3.3.2	C	3.3.3	A	3.4.1	80	3.4.2	511	3.4.3	C
3.5.1	B	3.5.2	N/A	3.5.3	D	3.5.4	N/A	3.5.5	C
3.5.6	B	3.5.7	B	3.5.8	B	3.5.9	B	3.5.10	A

3.5.11	B	3.5.12	C	3.5.13	D	3.5.14	C	3.5.15	110
3.5.16	A	3.5.17	B	3.5.18	B	3.5.19	64	3.5.20	B
3.5.21	B	3.5.22	B	3.5.23	B	3.5.24	D	3.5.25	C
3.5.26	D	3.5.27	C	3.5.28	A	3.5.29	D	3.5.30	C
3.5.31	B	3.6.1	False	3.6.2	False	3.6.3	N/A	3.6.4	N/A
3.6.5	N/A	3.6.6	N/A	3.6.7	B	3.6.8	144	3.6.9	N/A
3.6.10	N/A	3.6.11	N/A	3.6.12	N/A	3.6.13	N/A	3.6.14	N/A
3.6.15	B	3.6.16	N/A	3.6.17	B	3.6.18	C	3.6.19	N/A
3.6.20	N/A	3.6.21	C	3.6.22	D	3.6.23	D	3.6.24	N/A
3.6.25	B	3.6.26	D	3.6.27	D	3.6.28	C	3.6.29	B
3.6.30	A	3.6.31	C	3.6.32	A	3.6.33	B	3.6.34	A
3.6.35	1	3.6.36	A	3.6.37	19	3.6.38	199	3.6.39	C
3.6.40	4	3.6.41	4.25	3.6.42	1 : 1	3.6.43	D	3.6.44	B
3.6.45	D	3.6.46	C	3.6.47	B	3.6.48	D	3.6.49	B
3.6.50	A	3.7.1	A;C;D	3.8.1	B	3.8.2	D	3.8.3	B
3.8.4	12	3.8.5	C	3.9.1	B	3.9.2	C	3.9.3	N/A
3.9.4	N/A	3.9.5	C	3.9.6	B	3.9.7	C	3.9.8	C
3.9.9	C	3.9.10	A	3.9.11	A	3.9.12	B	3.9.13	80
3.9.14	C	3.9.15	B	3.9.16	D	3.10.1	N/A	3.10.2	C
3.10.3	N/A	3.10.4	B	3.10.5	D	3.10.6	A	3.10.7	D
3.10.8	A	3.10.9	D	3.10.10	A	3.10.11	B	3.10.12	C
3.10.13	D	3.10.14	B	3.10.15	A	3.10.16	B	3.10.17	A
3.10.18	D	3.10.19	B	3.10.20	8	3.10.21	A	3.10.22	B
3.10.23	C	3.10.24	B	3.11.1	N/A	3.11.2	A	3.11.3	N/A
3.12.1	B	3.12.2	N/A	3.12.3	N/A	3.12.4	B	3.12.5	D
3.12.6	C	3.12.7	N/A	3.12.8	N/A	3.12.9	N/A	3.12.10	D
3.12.11	B	3.12.12	A	3.12.13	D	3.12.14	B	3.12.15	D
3.12.16	C	3.12.17	B	3.12.18	C	3.12.19	A	3.12.20	B
3.13.1	D	3.14.1	N/A	3.14.2	N/A	3.14.3	A	3.14.4	B
3.14.5	A	3.14.6	A	3.14.7	A	3.14.8	A	3.14.9	256
3.14.10	B	3.14.11	B	3.14.12	B	3.15.1	B	3.15.2	B
3.15.3	B	3.15.4	N/A	3.15.5	C	3.15.6	D	3.15.7	A
3.15.8	B	3.15.9	A	3.15.10	C	3.15.11	15	3.15.12	C
3.15.13	86 : 86	3.15.14	D	3.15.15	C	3.15.16	B	3.16.1	N/A
3.16.2	A;D	3.16.3	N/A	3.16.4	B;C	3.16.5	A	3.16.6	N/A
3.16.7	D	3.16.8	D	3.16.9	C	3.16.10	C	3.16.11	A
3.16.12	D	3.16.13	18	3.16.14	B				

4 Programming and DS: Programming (108)

Programming in C. Recursion.

Mark Distribution in Previous GATE

Year	2021-1	2021-2	2020	2019	2018	2017-1	2017-2	2016-1	2016-2	Minimum	Average	Maximum
1 Mark Count	0	2	1	2	2	1	2	2	1	0	1.4	2
2 Marks Count	2	2	2	3	3	4	4	2	2	2	2.6	4
Total Marks	4	6	5	8	8	9	10	6	5	**4**	**6.7**	**10**

4.1 Aliasing (1) top

4.1.1 Aliasing: GATE CSE 2000 | Question: 1.16 top

https://gateoverflow.in/639

Aliasing in the context of programming languages refers to

A. multiple variables having the same memory location
B. multiple variables having the same value
C. multiple variables having the same identifier
D. multiple uses of the same variable

gate2000-cse programming easy aliasing

Answer

Answers: Aliasing

4.1.1 Aliasing: GATE CSE 2000 | Question: 1.16 top

https://gateoverflow.in/639

✔ Option is **A.**

In computer **programming**, **aliasing** refers to the situation where the **same memory location** can be accessed using different names. For instance, if a function takes two pointers A and B which have the same value, then the name **A aliases** the name **B**.

37 votes

-- Prasanna Ranganathan (3.9k points)

4.2 Arrays (11) top

4.2.1 Arrays: GATE CSE 2011 | Question: 22 top

https://gateoverflow.in/2124

What does the following fragment of C program print?

```
    char c[] = "GATE2011";
    char *p = c;
    printf("%s", p + p[3] - p[1]);
```

A. GATE2011
B. E2011
C. 2011
D. 011

gate2011-cse programming programming-in-c normal arrays

Answer

4.2.2 Arrays: GATE CSE 2015 Set 3 | Question: 30 top

https://gateoverflow.in/8486

Consider the following two C code segments. Y and X are one and two dimensional arrays of size n and $n \times n$ respectively, where $2 \leq n \leq 10$. Assume that in both code segments, elements of Y are initialized to 0 and each element $X[i][j]$ of array X is initialized to $i+j$. Further assume that when stored in main memory all elements of X are in same main memory page frame.

Code segment 1:

```
// initialize elements of Y to 0
// initialize elements of X[i][j] of X to i+j
for (i=0; i<n; i++)
    Y[i] += X[0][i];
```

Code segment 2:

```
// initialize elements of Y to 0
// initialize elements of X[i][j] of X to i+j
for (i=0; i<n; i++)
    Y[i] += X[i][0];
```

Which of the following statements is/are correct?

S1: Final contents of array Y will be same in both code segments

S2: Elements of array X accessed inside the for loop shown in code segment 1 are contiguous in main memory

S3: Elements of array X accessed inside the for loop shown in code segment 2 are contiguous in main memory

A. Only S2 is correct
B. Only S3 is correct
C. Only S1 and S2 are correct
D. Only S1 and S3 are correct

gate2015-cse-set3 programming-in-c normal arrays

Answer

4.2.3 Arrays: GATE CSE 2015 Set 3 | Question: 7 top https://gateoverflow.in/8401

Consider the following C program segment.

```
# include <stdio.h>
int main()
{
    char s1[7] = "1234", *p;
    p = s1 + 2;
    *p = '0';
    printf("%s", s1);
}
```

What will be printed by the program?

A. 12
B. 120400
C. 1204
D. 1034

gate2015-cse-set3 programming programming-in-c normal arrays

Answer

4.2.4 Arrays: GATE CSE 2017 Set 2 | Question: 55 top https://gateoverflow.in/118335

Consider the following C program.

```
#include<stdio.h>
#include<string.h>
int main() {
    char* c="GATECSIT2017";
    char* p=c;
    printf("%d", (int)strlen(c+2[p]-6[p]-1));
    return 0;
}
```

The output of the program is _______

gate2017-cse-set2 programming-in-c numerical-answers arrays

Answer

4.2.5 Arrays: GATE CSE 2019 | Question: 24 top https://gateoverflow.in/302824

Consider the following C program:

```
#include <stdio.h>
int main() {
        int arr[]={1, 2, 3, 4, 5, 6, 7, 8, 9, 0, 1, 2, 5}, *ip=arr+4;
        printf("%d\n", ip[1]);
        return 0;
```

```
}
```

The number that will be displayed on execution of the program is _______

gate2019-cse numerical-answers programming-in-c programming arrays

Answer

4.2.6 Arrays: GATE CSE 2020 | Question: 22 top

https://gateoverflow.in/333209

Consider the following C program.

```
#include <stdio.h>
int main ()  {
    int  a[4] [5] = {{1, 2, 3, 4, 5},
                    {6, 7,8, 9, 10},
                    {11, 12, 13, 14, 15},
                    {16, 17,18, 19, 20}};
    printf("%d\n", *(*(a+**a+2)+3));
    return(0);
}
```

The output of the program is _______.

gate2020-cse numerical-answers programming-in-c arrays

Answer

4.2.7 Arrays: GATE CSE 2021 Set 2 | Question: 10 top

https://gateoverflow.in/357530

Consider the following ANSI C program.

```
#include <stdio.h>
int main()
{
    int arr[4][5];
    int  i, j;
    for (i=0; i<4; i++)
          {
        for (j=0; j<5; j++)
        {
            arr[i][j] = 10 * i + j;
        }
    }
    printf("%d", *(arr[1]+9));
    return 0;
}
```

What is the output of the above program?

A. 14
B. 20
C. 24
D. 30

gate2021-cse-set2 programming-in-c arrays output

Answer

4.2.8 Arrays: GATE IT 2004 | Question: 58 top

https://gateoverflow.in/3701

Consider the following C program which is supposed to compute the transpose of a given 4×4 matrix M. Note that, there is an X in the program which indicates some missing statements. Choose the correct option to replace X in the program.

```
#include<stdio.h>
#define ROW 4
#define COL 4
int M[ROW][COL] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16};
main()
{
    int i, j, t;
    for (i = 0; i < 4; ++i)
    {
        X
    }
```

```
    for (i = 0; i < 4; ++i)
        for (j = 0; j < 4; ++j)
            printf ("%d", M[i][j]);
}
```

A.
```
for(j = 0; j < 4; ++j){
     t = M[i][j];
     M[i][j] = M[j][i];
     M[j][i] = t;
}
```

B.
```
for(j = 0; j < 4; ++j){
     M[i][j] = t;
     t = M[j][i];
     M[j][i] = M[i][j];
}
```

C.
```
for(j = i; j < 4; ++j){
     t = M[i][j];
     M[i][j] = M[j][i];
     M[j][i] = t;
}
```

D.
```
for(j = i; j < 4; ++j){
     M[i][j] = t;
     t = M[j][i];
     M[j][i] = M[i][j];
}
```

gate2004-it programming easy programming-in-c arrays

Answer

4.2.9 Arrays: GATE IT 2008 | Question: 49 top

https://gateoverflow.in/3359

What is the output printed by the following C code?

```
# include <stdio.h>
int main ()
{
    char a [6] = "world";
    int i, j;
    for (i = 0, j = 5; i < j; a [i++] = a [j--]);
    printf ("%s\n", a);
}
```

A. dlrow
B. Null string
C. dlrld
D. worow

gate2008-it programming programming-in-c normal arrays

Answer

4.2.10 Arrays: GATE IT 2008 | Question: 51 top

https://gateoverflow.in/3361

Consider the C program given below. What does it print?

```
#include <stdio.h>
int main ()
{
        int i, j;
        int a [8] = {1, 2, 3, 4, 5, 6, 7, 8};
        for(i = 0; i < 3; i++) {
             a[i] = a[i] + 1;
             i++;
        }
        i--;
        for (j = 7; j > 4; j--) {
              int i = j/2;
              a[i] = a[i] - 1;
        }
        printf ("%d, %d", i, a[i]);
}
```

A. $2, 3$
B. $2, 4$
C. $3, 2$
D. $3, 3$

gate2008-it programming programming-in-c normal arrays

Answer

4.2.11 Arrays: GATE IT 2008 | Question: 52 top

https://gateoverflow.in/3362

C program is given below:

```
# include <stdio.h>
int main ()
{
        int i, j;
        char a [2] [3] = {{'a', 'b', 'c'}, {'d', 'e', 'f'}};
        char b [3] [2];
        char *p = *b;
        for (i = 0; i < 2; i++) {
              for (j = 0; j < 3; j++) {
              *(p + 2*j + i) = a [i] [j];
              }
        }
}
```

What should be the contents of the array b at the end of the program?

A. a b
c d
e f

B. a d
b e
c f

C. a c
e b
d f

D. a e
d c
b f

gate2008-it programming programming-in-c normal arrays

Answer

Answers: Arrays

4.2.1 Arrays: GATE CSE 2011 | Question: 22 top

https://gateoverflow.in/2124

✔ **2011** is the answer.

In C, there is a rule that whatever character code be used by the compiler, codes of all alphabets and digits must be in order. So, if character code of 'A' is x, then for 'B' it must be $x + 1$.

Now %s means printf takes and address and prints all bytes starting from that address as characters till any byte becomes the code for '\0'. Now, the passed value to printf here is
$p + p[3] - p[1]$

p is the starting address of array c. $p[3] =' E'$ and $p[1] =' A'$. So, $p[3] - p[1] = 4$, and $p + 4$ will be pointing to the fifth position in the array c. So, printf starts printing from 2 and prints 2011.

(Here "$GATE2011$" is a string literal and by default a '\0' is added at the end of it by the compiler).

NB: In this question %s is not required.

```
printf(p + p[3] - p[1]);
```

Also gives the same result as first argument to printf is a character pointer and only if we want to pass more arguments we need to use a format string.

72 votes -- Arjun Suresh (330k points)

4.2.2 Arrays: GATE CSE 2015 Set 3 | Question: 30 top https://gateoverflow.in/8486

✔ Option is **C**. Only $S1$ and $S2$ are correct because Y have same element in both code and in code 1.

```
Y[i] += X[0][i];
```

This row major order (In C, arrays are stored in row-major order) which gives address of each element in sequential order $(1, 2, 3, \ldots, n)$ means we cross single element each time to move next shows contiguous in main memory but in code2 for:

$Y[i]+=X[i][0];$

```
We are crossing n element (row crossing with n element )to move next.
```

36 votes -- Anoop Sonkar (4.1k points)

4.2.3 Arrays: GATE CSE 2015 Set 3 | Question: 7 top https://gateoverflow.in/8401

✔
```
p = s1 + 2;
```

Type of s1 is char[7] and sizeof *s1 is sizeof (char) = 1. So, s1 + 2 will return address in s1 + 2 *sizeof(char) = address in s1 + 2. So, p now points to the third element in s1.

```
*p = '0';
```

The third element in s1 is made 0. So, 1234 becomes 1204. **C** choice.

57 votes -- Arjun Suresh (330k points)

4.2.4 Arrays: GATE CSE 2017 Set 2 | Question: 55 top https://gateoverflow.in/118335

✔
```
   char c[]="GATECSIT2017";
char *p=c;
printf("%d",strlen(c+2[p]-6[p]-1));
```

$2[p] = *(2+p) = p[2]$
$6[p] = *(6+p) = p[6]$
$c+2[p]-6[p]-1 = c+'T'-'I'-1 = c+11-1 = c+10$ (In any character coding all alphabet letters are assigned consecutive int values as per C)

printf will print 2 which is the length of " 17".

71 votes -- Arjun Suresh (330k points)

4.2.5 Arrays: GATE CSE 2019 | Question: 24 top https://gateoverflow.in/302824

✔ 6

ip is an integer pointer and the initial assignment sets it to the element at array index 4 i.e. 5.(holds address of ar index 4)
The next statement refers to the next integer after it which is $6(ip[1] = *(ip+1))$.

19 votes -- vin101 (853 points)

4.2.6 Arrays: GATE CSE 2020 | Question: 22 top https://gateoverflow.in/333209

✔ 'a' is a two dimensional array.

main

array

a

0,0 int 1	0,1 int 2	0,2 int 3	0,3 int 4	0,4 int 5
1,0 int 6	1,1 int 7	1,2 int 8	1,3 int 9	1,4 int 10
2,0 int 11	2,1 int 12	2,2 int 13	2,3 int 14	2,4 int 15
3,0 int 16	3,1 int 17	3,2 int 18	3,3 int 19	3,4 int 20

- $a =$ address of 0^{th} index of 2-D array which means address of 1-D array
- $^*a =$ address of 0^{th} index element of 0^{th} index 1-D array
- $^{**}a =$ value at 0^{th} index element of 0^{th} index 1-D array
  - $\implies$ $^{**}a = 1$
  - $\implies$ $^{**}a + 2 = 1 + 2 = 3$
- $a + 3 =$ address of 3^{rd} index 1-D array
- $^*(a + 3) =$ address of 0^{th} index element of 3^{rd} index 1-D array
- $^*(a + 3) + 3 =$ address of 3^{rd} index element of 3^{rd} index 1-D array
- $^*(^*(a + 3) + 3) =$ value at 3^{rd} index element of 3^{rd} index 1-D array $= 19$

Correct Answer: 19.

28 votes -- Shaik Masthan (50.4k points)

4.2.7 Arrays: GATE CSE 2021 Set 2 | Question: 10 top https://gateoverflow.in/357530

✔ $arr[1] + 9$

$arr[1]$ is a pointer to 1D array of 5 integers and so the above expression involves pointer arithmetic.

For a pointer value p and integer value d,

- $p + d \implies p + sizeof(*p) + d$

So,

```
arr[1]+9 = *(arr+1) + 9 //arr is again a pointer but to the 2D array arr[4][5]

arr+1 = arr + sizeof(*arr)
      = arr + 5 * sizeof (int)

*(arr+1)+9 = arr + 5 * sizeof(int) + sizeof(**arr) * 9
//* operator in *(arr+1) just changes the type of the pointer here
        = arr + 5 * sizeof(int) + 9 * sizeof(int)
        = arr + 14 * sizeof(int)
```

Now, C language follows row major ordering for arrays which means that when a multi dimensional array gets linearized in memory the lower dimensions get arranged contiguously. For the 2D array $arr[4][5]$ it'll be

5 elements of $arr[0][5]$ followed by 5 elements of $arr[1][5]$ followed by 5 elements of $arr[2][5]$ and so on.

So, the 14^{th} element will be at row number $\lceil 14/5 \rceil = 2$ and column number $14\%5 = 4$, which is $arr[2][4] = 10 \times 2 + 4 = 24$.

More Read: https://gatecse.in/chapter-3-pointers/

References

1 votes -- gatecse (62.6k points)

4.2.8 Arrays: GATE IT 2004 | Question: 58 top https://gateoverflow.in/3701

✔ Ooption **C**:

look at the initial value of j, if j starts with 0, then double *for* loop will swap $M[i][j]$ with $M[j][i]$ and also $M[j][i]$ and $M[i][j]$ so the matrix M will remain unchanged, so to avoid this double swapping we need to initialize $j = i$ and swap only upper triangular matrix with lower triangular matrix.

```
for(j = i; j < 4; ++j){
 // code for swapping M[i][j] with M[j][i]
     t = M[i][j];
     M[i][j] = M[j][i];
     M[j][i] = t;
}
```

45 votes -- Vikrant Singh (11.2k points)

4.2.9 Arrays: GATE IT 2008 | Question: 49 top https://gateoverflow.in/3359

✔ Char $a[6] =$

w	o	r	l	d	\0

After the loop executes for the first time,

$a[0] = a[5]$

$a[0] =$`\0`

Next two more iterations of the loop till $i < j$ condition becomes false, are not important for the output as the first position is '\0';

printf(‘’%s’’, a);

printf function for format specifier '%s' prints the characters from the corresponding parameter (which should be an address) until "\0" occurs. Here, first character at a is "\0" and hence it will print NOTHING.

So, option (**B**).

33 votes -- Mitali (151 points)

4.2.10 Arrays: GATE IT 2008 | Question: 51 top https://gateoverflow.in/3361

✔ Answer is (**C**) $3, 2$

First 2 variable integer type declared named i and j

Then **int type array** $a[8]$ declared and initialized.

$a[0] = 1, a[1] = 2, a[2] = 3, a[3] = 4, a[4] = 5, a[5] = 6, a[6] = 7, a[7] = 8$

Then for loop started

$i = 0, i < 3$ (true)

$a[0] = a[0] + 1 = 1 + 1 = 2$

$i++$ (outside for loop) , $i++$ (inside for loop);

$i = 2, i < 3$ (true)

$a[2] = a[2] + 1 = 3 + 1 = 4$

$i++, i++$(outside for loop) ,

$i = 4, i < 3$ (false) **//Now come out of loop**

$i--;$ (so $i = 3$)

Now another for loop started where in loop integer type variable named i declared

Block Scope: A Block is a set of statements enclosed within left and right braces ({ and } respectively). Blocks may be nested in C (a block may contain other blocks inside it). A variable declared in a block is accessible in the block and all inner blocks of that block, but not accessible outside the block.

What if the inner block itself has one variable with the same name?
If an inner block declares a variable with the same name as the variable declared by the outer block, then the visibility of the outer block variable ends at the point of declaration by inner block.

So here inner block **int i** has the scope in this block only and outer block int i visibility is not allowed in that block

$j = 7, j > 4$(true)

int $i = 7/2 = 3$

$a[3] = a[3] - 1 = 4 - 1 = 3$

$j = 6, j > 4$ (true)

int $i = 6/2 = 3$

$a[3] = a[3] - 1 = 3 - 1 = 2$

$j = 5, j > 4$ (true)

int $i = 5/2 = 2$

$a[2] = a[2] - 1 = 4 - 1 = 3$

$j = 4, j > 4$ (false)

Now when the for loop ends its variable named i scope is also end and the outer block variable now visible. So, in printf outer variable i is used.

So, the output would be: $3, 2$.

124 votes -- Kalpna Bhargav (2.5k points)

4.2.11 Arrays: GATE IT 2008 | Question: 52 top

https://gateoverflow.in/3362

✔ The correct answer is option ($\mathbf{B}$).

first integer type two variables declared i and j

then an integer type $2-d$ array $a[2][3]$ is declared and initialized and $2-d$ array $b[3][2]$ is created but not initialized. i.e

	address	value		address	value
$a[0]0[]$	2000	a	$b[0][0]$	3000	garbage value
$a[0][1]$	2001	b	$b[0][1]$	3001	garbage value
$a[0][2]$	2002	c	$b[1][0]$	3002	garbage value
$a[1][0]$	2003	d	$b[1][1]$	3003	garbage value
$a[1][1]$	2004	e	$b[2][0]$	3004	garbage value
$a[1][2]$	2005	f	$b[2][1]$	3005	garbage value

now the char type pointer is declared and the base address of array b is put in it. so $p = 3000$

now the for loop is started where i is initialized to 0 ,so

$i = 0 : i < 2$ (true)

$j = 0; j < 3$ (true)

$*(3000 + 2 * 0 + 0) = a[0][0] \Rightarrow *(3000) = a$

$j++$

$j = 1; j < 3$ (true)

$*(3000 + 2 * 1 + 0) = a[0][1] \Rightarrow *(3002) = b$

$j++$

$j = 2; j < 3$ (true)

$*(3000 + 2 * 2 + 0) = a[0][2] \Rightarrow *(3004) = c$

$j++$

$j = 3; j < 3$ (false)

$i++$

$i = 1 : i < 2$ (true)

$j = 0; j < 3$ (true)

$*(3000 + 2 * 0 + 1) = a[1][0] \Rightarrow *(3001) = d$

$j++$

$j = 1; j < 3$ (true)

$*(3000 + 2 * 1 + 1) = a[1][1] \Rightarrow *(3003) = e$

$j++$

$j = 2; j < 3$ (true)

$*(3000 + 2 * 2 + 1) = a[1][2] \Rightarrow *(3005) = f$

$j++$

$j = 3; j < 3$ (false)

$i + +$

now the values in array b is

$b[0][0]$	3000	a
$b[0][1]$	3001	d
$b[1][0]$	3002	b
$b[1][1]$	3003	e
$b[2][0]$	3004	c
$b[2][1]$	3005	f

Hence, the output will be (**B**) choice.

Note:

```
*(p + 2*j + i)
```

$p+$ size of inner dimension $*j + i$, hence is same as $p[j][i]$. Hence with this statement we can identify that the code is transposing the matrix a and storing in b using pointer p.

47 votes -- Kalpna Bhargav (2.5k points)

4.3 Goto (2) top

4.3.1 Goto: GATE CSE 1989 | Question: 3-i top https://gateoverflow.in/87095

An unrestricted use of the "go to" statement is harmful because of which of the following reason (s):

A. It makes it more difficult to verify programs.
B. It makes programs more inefficient.
C. It makes it more difficult to modify existing programs.
D. It results in the compiler generating longer machine code.

gate1989 normal programming goto

Answer

4.3.2 Goto: GATE CSE 1994 | Question: 1.5 top https://gateoverflow.in/2442

An unrestricted use of the "*goto*" statement is harmful because

A. it makes it more difficult to verify programs
B. it increases the running time of the programs
C. it increases the memory required for the programs
D. it results in the compiler generating longer machine code

gate1994 programming easy goto

Answer

Answers: Goto

4.3.1 Goto: GATE CSE 1989 | Question: 3-i top https://gateoverflow.in/87095

✔ **Solution :** A) It makes it more difficult to verify programs.

Proof of correctness : https://en.wikipedia.org/wiki/Goto#Criticism
(1st Paragraph, last 4 lines.)

Option B: goto has no role in making a program inefficient. Adding a "goto" introduces a branch instruction but assuming same program logic this can never be avoided and even when replaced by a proper structural construct like for/while/if -- will still have the branch instruction.

Option C: Actually using "goto" kind of makes it easy to modify an existing program as to make the control flow from say line number 'x' to line number 'y', we can simply do a "goto". This is much harder to do in a proper structural way.

Option D: This is also false as goto introduces only a single "JUMP" instruction (machine code will be its equivalent; OPCODE for JMP followed by the destination address) and equivalent structural constructs will have longer machine codes.

References

17 votes -- Siddharth Mahapatra (1.2k points)

4.3.2 Goto: GATE CSE 1994 | Question: 1.5 top

https://gateoverflow.in/2442

✔ Use of *goto* takes out the structural decomposition of the code and hence it becomes very difficult to verify or debug the code. As far as performance or memory impact is concerned, *goto* has no effect on them.

Correct Answer: A

37 votes -- Arjun Suresh (330k points)

4.4 Identify Function (4) top

4.4.1 Identify Function: GATE CSE 1995 | Question: 3 top

https://gateoverflow.in/2639

Consider the following high level programming segment. Give the contents of the memory locations for variables W, X, Y and Z after the execution of the program segment. The values of the variables A and B are $5CH$ and $92H$, respectively. Also indicate error conditions if any.

```
var
    A, B, W, X, Y   :unsigned byte;
    Z               :unsigned integer, (each integer is represented by two bytes)
begin
    X               :=A+B
    Y               :=abs(A-B);
    W               :=A-B
    Z               :=A*B
end;
```

gate1995 programming identify-function descriptive

Answer

4.4.2 Identify Function: GATE CSE 1998 | Question: 2.13 top

https://gateoverflow.in/1685

What is the result of the following program?

```
    program side-effect (input, output);
    var x, result: integer;
    function f (var x:integer):integer;
    begin
        x:x+1;f:=x;
    end
begin
    x:=5;
    result:=f(x)*f(x);
    writeln(result);
    end
```

A. 5
B. 25
C. 36
D. 42

gate1998 programming normal identify-function

Answer

4.4.3 Identify Function: GATE CSE 2017 Set 2 | Question: 43 top

https://gateoverflow.in/118388

Consider the following snippet of a C program. Assume that swap $(\&x, \&y)$ exchanges the content of x and y:

```
int main () {
    int array[] = {3, 5, 1, 4, 6, 2};
    int done =0;
    int i;
    while (done==0) {
        done =1;
```

```
        for (i=0; i<=4; i++) {
            if (array[i] < array[i+1]) {
                swap(&array[i], &array[i+1]);
                done=0;
            }
        }
        for (i=5; i>=1; i--) {
            if (array[i] > array[i-1]) {
                swap(&array[i], &array[i-1]);
                done =0;
            }
        }
    }
    printf("%d", array[3]);
}
```

The output of the program is _______

gate2017-cse-set2 programming algorithms numerical-answers identify-function

Answer

4.4.4 Identify Function: GATE IT 2004 | Question: 15 top

https://gateoverflow.in/3656

Let x be an integer which can take a value of 0 or 1. The statement

```
if (x == 0) x = 1; else x = 0;
```

is equivalent to which one of the following ?

A. $x = 1 + x;$
B. $x = 1 - x;$
C. $x = x - 1;$
D. $x = 1\%x;$

gate2004-it programming easy identify-function

Answer

Answers: Identify Function

4.4.1 Identify Function: GATE CSE 1995 | Question: 3 top

https://gateoverflow.in/2639

✔ The maximum value that can be accommodated in an unsigned byte $= 255$ and unsigned int $= 65535$.

A and B are given in Hexadecimal.

- $A = 5C_H = (92)_{10}$
- $B = 92_H = (146)_{10}$
- $X = A + B = (238)_{10} = EE_H$
- $Y = \text{abs}(A - B) = (54)_{10} = 36_H$
- $W = A - B = (-54)_{10}$

Negative numbers represented in 2's complement form $\implies -54 = 11001010$ (in 8-bit representations)

But W is unsigned, therefore it cannot look for the sign $\implies W = 11001010 = CA_H$
$Z = A * B = (13432)_{10} = 3478_H$

26 votes -- Ravi Ranjan (3k points)

4.4.2 Identify Function: GATE CSE 1998 | Question: 2.13 top

https://gateoverflow.in/1685

✔ Call by value: 36,
Call by reference: undefined behaviour for C/C++ but 42 for languages having $*$ as a sequence point.

$f(x) * f(x);$

If the value of x is being modified inside the function (call by reference) we cannot be sure if this modified value or the old value will be passed as argument for the second call to $f()$. This is because left and right operand of any arithmetic expression in C/C++ can be evaluated in any order. For languages like Java, strict left-right order is maintained.

18 votes -- Arjun Suresh (330k points)

4.4.3 Identify Function: GATE CSE 2017 Set 2 | Question: 43 top https://gateoverflow.in/118388

✔ Well, the above program is sorting the array in descending order.

Initially, *while loop* starts execution by *evaluating the iniatial condition*

> while(done==0)

For the first time the first for loop will be executed completey, the content of array will be as follows :

$5, 3, 4, 6, 2, 1$

After the second for executed completey the content of array will be as follows:

$6, 5, 3, 4, 2, 1$

But the value variable done is still 0 so while loop will execute again,so now the content of array after executing the first for loop will be $6, 5, 4, 3, 2, 1$ and no change in second for loop but still the done variable is 0.

So, while loop execute again,now done variable is modified to 1 and there will be no change in done variable because inside first and second for loop no if condition will satisfied .

Finally, the while condition is evaluted false and value of $array[3]$ will be printed which is 3.

54 votes -- Manoj Kumar (26.7k points)

4.4.4 Identify Function: GATE IT 2004 | Question: 15 top https://gateoverflow.in/3656

✔ Firstly, our requirement is for $x = 1$ it makes '0' and for $x = 0$ it makes '1'

Let's consider options one by one:

A. $x = 1 + x$
   - For $x = 1$, it gives 2 So, False

B. $x = 1 - x$
   - Here, **B** is correct, as
     - For $x = 0$, it gives 1.
     - For $x = 1$, it gives 0.

C. $x = x - 1$
   - For $x = 0$, it gives -1. So, False

D. $x = 1\%x$
   - For $x = 0$, it gives $1\%0$. I think it is undefined
   - Even if we consider $x = x\%1$
     - for $x = 0$,it gives $0\%1 = 0$ But we require 1.

So, **Option (B)** is correct.

31 votes -- Himanshu Agarwal (12.4k points)

4.5 Loop Invariants (8) top

4.5.1 Loop Invariants: GATE CSE 1987 | Question: 7a top https://gateoverflow.in/82425

List the invariant assertions at points A, B, C, D and E in program given below:

```
Program division (input, output)
Const
    dividend = 81;
    divisor = 9;
Var  remainder, quotient:interger
```

```
begin
    (*(dividend >= 0) AND (divisor > 0)*)
    remainder := dividend;
    quotient := 9;
    (*A*)
While (remainder >= 0) do
begin (*B*)
    quotient := quotient + 1;
    remainder := remainder - divisor;
    (*C*)
end;
    (*D*)
    quotient := quotient - 1;
    remainder := remainder + divisor;
    (*E*)
end
```

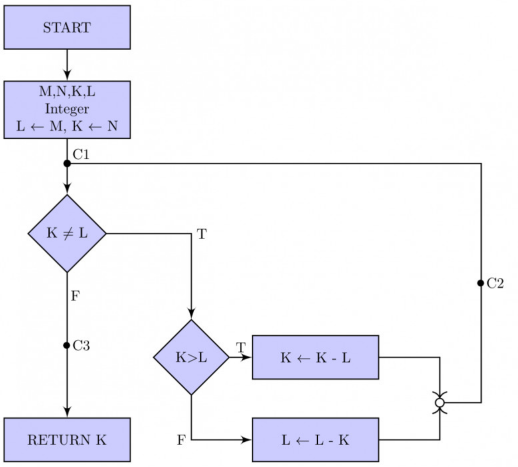
gate1987 programming loop-invariants descriptive

Answer

4.5.2 Loop Invariants: GATE CSE 1988 | Question: 6ii top

☛ https://gateoverflow.in/94364

Below figure is the flow-chart corresponding to a program to calculate the gcd of two integers, M and N respectively, $(M, N > 0)$. Use assertions at the cut point C_1, C_2 and C_3 to prove that the flow-chart is correct.



gate1988 normal descriptive loop-invariants

Answer

4.5.3 Loop Invariants: GATE CSE 1988 | Question: 8ii top

☛ https://gateoverflow.in/94379

Consider the two program segments below:

a.
```
for
    i:=1 to f(x) by 1 do
    S
end
```

b.
```
i:=1;
While    i<=f(x) do
         S
         i:=i+1
end
```

Under what conditions are these two programs equivalent? Treat S as any sequence of statements and f as a function.

gate1988 programming descriptive loop-invariants

Answer

4.5.4 Loop Invariants: GATE CSE 1991 | Question: 1,vi top

☛ https://gateoverflow.in/504

Consider the following PASCAL program segment:

```
if i mod 2 = 0 then
    while i >= 0 do
    begin
        i := i div 2;
        if i mod 2 < > 0  then i := i - 1;
        else i := i - 2;
    end;
```

An appropriate loop-invariant for the while-loop is ________

gate1991 programming loop-invariants normal fill-in-the-blanks

Answer

4.5.5 Loop Invariants: GATE CSE 2004 | Question: 32 top

https://gateoverflow.in/1029

Consider the following program fragment for reversing the digits in a given integer to obtain a new integer. Let $n = d_1 \, d_2 \, \ldots \, d_m$.

```
int n, rev;
rev = 0;
while(n > 0) {
    rev = rev * 10 + n%10;
    n = n/10;
}
```

The loop invariant condition at the end of the i^{th} iteration is:

A. $n = d_1 \, d_2 \, \ldots \, d_{m-i} \quad$ **and** $\quad \text{rev} = d_m \, d_{m-1} \, \ldots \, d_{m-i+1}$
B. $n = d_{m-i+1} \, \ldots \, d_{m-1} \, d_m \quad$ **or** $\quad \text{rev} = d_{m-i} \, \ldots \, d_2 \, d_1$
C. $n \neq \text{rev}$
D. $n = d_1 \, d_2 \, \ldots \, d_m \quad$ **or** $\quad \text{rev} = d_m \, \ldots \, d_2 \, d_1$

gate2004-cse programming loop-invariants normal

Answer

4.5.6 Loop Invariants: GATE CSE 2015 Set 1 | Question: 33 top

https://gateoverflow.in/8276

Consider the following pseudo code, where x and y are positive integers.

```
begin
    q := 0
    r := x
   while r ≥ y do
      begin
      r := r - y
      q := q + 1
    end
end
```

The post condition that needs to be satisfied after the program terminates is

A. $\{r = qx + y \wedge r < y\}$
B. $\{x = qy + r \wedge r < y\}$
C. $\{y = qx + r \wedge 0 < r < y\}$
D. $\{q + 1 < r - y \wedge y > 0\}$

gate2015-cse-set1 programming loop-invariants normal

Answer

4.5.7 Loop Invariants: GATE CSE 2016 Set 2 | Question: 35 top

https://gateoverflow.in/39578

The following function computes X^Y for positive integers X and Y.

```
int exp (int X, int Y) {
     int res =1, a = X, b = Y;

     while (b != 0) {
         if (b % 2 == 0) {a = a * a; b = b/2; }
         else          {res = res * a; b = b - 1; }
```

```
    }
    return res;
}
```

Which one of the following conditions is TRUE before every iteration of the loop?

A. $X^Y = a^b$
B. $(res * a)^Y = (res * X)^b$
C. $X^Y = res * a^b$
D. $X^Y = (res * a)^b$

gate2016-cse-set2 programming loop-invariants normal

Answer

4.5.8 Loop Invariants: GATE CSE 2017 Set 2 | Question: 37 top

https://gateoverflow.in/118381

Consider the C program fragment below which is meant to divide x by y using repeated subtractions. The variables x, y, q and r are all unsigned int.

```
while (r >= y) {
    r=r-y;
    q=q+1;
}
```

Which of the following conditions on the variables x, y, q and r before the execution of the fragment will ensure that the loop terminated in a state satisfying the condition $x == (y * q + r)$?

A. $(q == r)$ && $(r == 0)$
B. $(x > 0)$ && $(r == x)$ && $(y > 0)$
C. $(q == 0)$ && $(r == x)$ && $(y > 0)$
D. $(q == 0)$ && $(y > 0)$

gate2017-cse-set2 programming loop-invariants

Answer

Answers: Loop Invariants

4.5.1 Loop Invariants: GATE CSE 1987 | Question: 7a top

https://gateoverflow.in/82425

A: remainder >= 0 and quotient = 9;

B: remainder >= 0 and quotient <= divident/divisor

C: quotient - 1 <= divident/divisor

D: remainder < 0 and quotient - 1 = ⌊ divident/divisor ⌋

E: divident = divisor * quotient + remainder

PS: To be fixed.

6 votes

-- Arjun Suresh (330k points)

4.5.2 Loop Invariants: GATE CSE 1988 | Question: 6ii top

https://gateoverflow.in/94364

Cond 3 : Evaluating condition where two integers are equal and thus GCD(x,x)=x Thus it is returning X itself.
Cond 2 : We are reducing the Greater integer among 2 by the smallest one which will ultimately reduce it by factor of smaller integer.
Cond1: Provides updated value at other iteration and Original value at 1st iteration./

EX: M=9 N=6
L=9,K=6
Iteration 1 ::Cond 1 :(9! = 6)True --> (K>L)False --> L=3 -->Cond2::Cond1 (Updated value : L=3 K=6)
Iteration 2 ::Cond1 :(3!=6) True --> (K>L)True --> K=3--> Cond2:: Cond1(Updated Value L=3,K=3)
Iteration 3 ::Cond1 (3!=6) False Cond 3 :: Return K=3 **Which is GCD(9,6)=3**

8 votes -- Akshay Saxena (8.3k points)

4.5.3 Loop Invariants: GATE CSE 1988 | Question: 8ii top https://gateoverflow.in/94379

In both program segment if f(x) returns same value

1 votes -- learner_geek (1.1k points)

4.5.4 Loop Invariants: GATE CSE 1991 | Question: 1,vi top https://gateoverflow.in/504

✔ **Loop invariant** is some condition which holds at the end of each iteration of the loop. i.e. it is *"invariant"* => does not vary (or change). It might change inside one iteration, but it will be true at the end of every iteration.

We often use loop invariants to prove that our algorithm works correctly.

In the given program, a loop invariant is:

```
i mod 2 = 0
```

i.e. **i is even after every iteration**.

One can verify this as follows:

- Before the execution of first iteration the loop invariant is true, because of this line of code:

  ```
  if i mod 2 = 0 then
  ```

- In every iteration, we divide i by 2, so now i will be either odd or even.
  - If odd, we subtract 1 from i

    ```
    if i mod 2 < > 0  then i := i - 1;
    ```

  - so it's now even.
  - otherwise, if even, we subtract 2 from i

    ```
    else i := i - 2;
    ```

  - so, it remains even
- So, at the end of every iteration *i remains even.*

https://stackoverflow.com/questions/3221577/what-is-a-loop-invariant

References

22 votes -- Rishabh Gupta (12.5k points)

4.5.5 Loop Invariants: GATE CSE 2004 | Question: 32 top https://gateoverflow.in/1029

✔ A loop invariant is something that hold at the start of a loop, across each iteration (inside an iteration it can change but before the iteration ends original condition must be true) and at the end also. So, we can check for the satisfiability of the condition at the loop header for start of the loop, for each iteration and also at the exit.

Here, in each iteration the right most digit of n, is moving to the right end of rev. So, answer is (**A**). i.e. the 2 conditions given in (A) choice are true on entry to loop, after each iteration (not necessarily during an iteration), and at end of loop.

32 votes -- Arjun Suresh (330k points)

4.5.6 Loop Invariants: GATE CSE 2015 Set 1 | Question: 33 top https://gateoverflow.in/8276

✔ Correct Option: **B**

The loop terminates when $r < y$. So, $r < y$ is one post condition.

In each iteration q is incremented by 1 and y is subtracted from r. Initial value of r is x. So, loop iterates x/y times and q will be equal to x/y and $r = x\%y \Rightarrow x = qy + r$;

60 votes -- Arjun Suresh (330k points)

4.5.7 Loop Invariants: GATE CSE 2016 Set 2 | Question: 35 top https://gateoverflow.in/39578

✔ Take $X = 10, Y = 3$

In that case,

Before Iteration 1:

$res = 1, a = 10, b = 3$

All options are satisfied here.

Iteration 1:

- $while(b! = 0) \implies 3! = 0 \quad \checkmark$
- $if(b\%2 == 0) \implies 3\%2 == 0 \quad \times$
- $else$

$$res = res * a \implies res = 1 * 10 = 10$$

$$b = b - 1 \implies b = 3 - 1 = 2$$

Before Iteration 2:

$res = 10, a = 10, b = 2$

option **A** : $X^Y = a^b \implies 10^3 = 10^2 \quad \times$

option **B** : $(res * a)^Y = (res * X)^b \implies (10 * 10)^3 = (10 * 10)^2 \quad \times$

option **C** : $X^Y = res * a^b \implies 10^3 = 10 * 10^2 \quad \checkmark$

option **D** : $X^Y = (res * a)^b \implies 10^3 = (10 * 10)^2 \quad \times$

Lets see one more iteration to verify option **C**.

Iteration 2:

$res = 10, a = 10, b = 2$

- $while(b! = 0) \implies 2! = 0 \quad \checkmark$
- $if(b\%2 == 0) \implies 2\%2 == 0 \quad \checkmark$

$$a = a * a \\ = 10 * 10 = 100$$

$$b = \frac{b}{2} \\ = \frac{2}{2} = 1$$

Before Iteration 3:

$res = 10, a = 100, b = 1$

Option **C** : $X^Y = res * a^b \implies 10^3 = 10 * 100^1 = 10^3 \quad \checkmark$

Option C is answer

64 votes -- Akash Kanase (36k points)

4.5.8 Loop Invariants: GATE CSE 2017 Set 2 | Question: 37 top https://gateoverflow.in/118381

✔ Here, $x == (y * q + r)$ says $q =$ quotient and $r =$ remainder.

To divide a number with repeated subtraction, quotient should be initialized to 0 and should be incremented for each subtraction.

Initially $q = 0 \Rightarrow r = x$.

$\therefore$ Initial conditions should be **C**] $(q == 0)$ && $(r == x)$ && $(y > 0)$.

40 votes -- Kantikumar (3.4k points)

4.6 Parameter Passing (12) top

4.6.1 Parameter Passing: GATE CSE 1992 | Question: 10b top https://gateoverflow.in/43584

Show the activation records and the display structure just after the procedures called at lines marked x and y have started their execution. Be sure to indicate which of the two procedures named A you are referring to.

```
Program Test;
    Procedure A;
        Procedure B;
            Procedure A;
            begin
            ……
            end A;
        begin
            y: A;
        end B;
    begin
        B;
    end A;

begin
    x: A;
end Test
```

gate1992 parameter-passing programming runtime-environments normal descriptive

Answer

4.6.2 Parameter Passing: GATE CSE 1994 | Question: 1.20 top

https://gateoverflow.in/305

In which of the following cases is it possible to obtain different results for call-by-reference and call-by-name parameter passing methods?

A. Passing a constant value as a parameter
B. Passing the address of an array as a parameter
C. Passing an array element as a parameter
D. Passing an array

gate1994 programming parameter-passing easy

Answer

4.6.3 Parameter Passing: GATE CSE 2001 | Question: 2.17 | UGCNET-AUG2016-III: 21 top

What is printed by the print statements in the program $P1$ assuming call by reference parameter passing?

```
Program P1()
{
    x = 10;
    y = 3;
    func1(y,x,x);
    print x;
    print y;
}

func1(x,y,z)
{
    y = y + 4;
    z = x + y + z
}
```

A. 10, 3
B. 31, 3
C. 27, 7
D. None of the above

gate2001-cse programming compiler-design parameter-passing normal runtime-environments ugcnetaug2016iii

Answer

4.6.4 Parameter Passing: GATE CSE 2003 | Question: 73 top

https://gateoverflow.in/960

The following program fragment is written in a programming language that allows global variables and does not allow nested declarations of functions.

```
global int i=100, j=5;
void P(x) {
 int i=10;
    print(x+10);
```

GOLD Q. MUST READ DEEPAK POONIA SIR'S ANSWERS

```
    i=200;
    j=20;
    print (x);
}
main() {P(i+j);}
```

If the programming language uses static scoping and call by need parameter passing mechanism, the values printed by the above program are:

A. 115, 220
B. 25, 220
C. 25, 15
D. 115, 105

gate2003-cse compiler-design normal runtime-environments parameter-passing

Answer

4.6.5 Parameter Passing: GATE CSE 2008 | Question: 60 top https://gateoverflow.in/483

What is printed by the following C program?

```
int f(int x, int *py, int **ppz)
{
    int y, z;
    **ppz += 1; z = **ppz;    // corrected z = *ppz; to z = **ppz;
    *py += 2; y = *py;
    x += 3;
    return x+y+z;
}

void main()
{
    int c, *b, **a;
    c = 4; b = &c; a = &b;
    printf("%d", f(c, b, a));

}
```

A. 18
B. 19
C. 21
D. 22

gate2008-cse programming programming-in-c normal parameter-passing

Answer

4.6.6 Parameter Passing: GATE CSE 2010 | Question: 11 top https://gateoverflow.in/2184

What does the following program print?

```
#include<stdio.h>

void f(int *p, int *q) {
    p=q;
    *p=2;
}

int i=0, j=1;

int main() {
    f(&i, &j);
    printf("%d %d\n", i,j);
    return 0;
}
```

A. 2 2
B. 2 1
C. 0 1
D. 0 2

gate2010-cse programming programming-in-c easy parameter-passing

Answer

4.6.7 Parameter Passing: GATE CSE 2013 | Question: 42 top

https://gateoverflow.in/60

What is the return value of $f(p,p)$, if the value of p is initialized to 5 before the call? Note that the first parameter is passed by reference, whereas the second parameter is passed by value.

```
int f (int &x, int c) {
       c = c - 1;
       if (c==0) return 1;
       x = x + 1;
       return f(x,c) * x;
}
```

gate2013-cse compiler-design normal marks-to-all numerical-answers parameter-passing runtime-environments

Answer

4.6.8 Parameter Passing: GATE CSE 2016 Set 1 | Question: 15 top

https://gateoverflow.in/39642

Consider the following C program.

```
# include <stdio.h>
void mystery (int *ptra, int *ptrb) {
     int *temp;
     temp = ptrb;
     ptrb =ptra;
     ptra = temp;
}
int main () {
    int a = 2016, b=0, c= 4, d = 42;
    mystery (&a, &b);
    if (a < c)
          mystery (&c, &a);
    mystery (&a, &d);
    printf("%d\n", a);
}
```

The output of the program is _________.

gate2016-cse-set1 programming-in-c easy numerical-answers parameter-passing

Answer

4.6.9 Parameter Passing: GATE CSE 2016 Set 2 | Question: 12 top

https://gateoverflow.in/39565

The value printed by the following program is _______.

```
      void f (int * p, int m) {
           m = m + 5;
          *p = *p + m;
           return;
      }
      void main () {
       int i=5, j=10;

       f (&i, j);
       printf ("%d", i+j);
     }
```

gate2016-cse-set2 programming-in-c normal numerical-answers parameter-passing

Answer

4.6.10 Parameter Passing: GATE CSE 2018 | Question: 29 top

https://gateoverflow.in/204103

```
#include<stdio.h>
void fun1(char* s1, char* s2){
    char* temp;
    temp = s1;
    s1 = s2;
    s2 = temp;
}
void fun2(char** s1, char** s2){
```

```
    char* temp;
    temp = *s1;
    *s1 = *s2;
    *s2 = temp;
}
int main(){
    char *str1="Hi", *str2 = "Bye";
    fun1(str1, str2); printf("%s %s", str1, str2);
    fun2(&str1, &str2); printf("%s %s", str1, str2);
    return 0;
}
```

The output of the program above is:

A. Hi Bye Bye Hi
B. Hi Bye Hi Bye
C. Bye Hi Hi Bye
D. Bye Hi Bye Hi

gate2018-cse programming-in-c pointers parameter-passing normal programming

Answer

4.6.11 Parameter Passing: GATE IT 2006 | Question: 50 top

https://gateoverflow.in/3593

Which one of the choices given below would be printed when the following program is executed?

```
#include <stdio.h>
void swap (int *x, int *y)
{
    static int *temp;
    temp = x;
    x = y;
    y = temp;
}
void printab ()
{
    static int i, a = -3, b = -6;
    i = 0;
    while (i <= 4)
    {
        if ((i++)%2 == 1) continue;
        a = a + i;
        b = b + i;
    }
    swap (&a, &b);
    printf("a =  %d, b = %d\n", a, b);
}
main()
{
    printab();
    printab();
}
```

A. $a = 0, b = 3$
   $a = 0, b = 3$
B. $a = 3, b = 0$
   $a = 12, b = 9$
C. $a = 3, b = 6$
   $a = 3, b = 6$
D. $a = 6, b = 3$
   $a = 15, b = 12$

gate2006-it programming programming-in-c normal parameter-passing

Answer

4.6.12 Parameter Passing: GATE IT 2008 | Question: 50 top

https://gateoverflow.in/3360

Consider the C program below. What does it print?

```
# include <stdio.h>
# define swapl (a, b) tmp = a; a = b; b = tmp
void swap2 ( int a, int b)
{
```

```
        int tmp;
        tmp = a; a = b; b = tmp;
 }
void swap3 (int*a, int*b)
{
        int tmp;
        tmp = *a; *a = *b; *b = tmp;
}
int main ()
{
        int num1 = 5, num2 = 4, tmp;
        if (num1 < num2) {swap1 (num1, num2);}
        if (num1 < num2) {swap2 (num1 + 1, num2);}
        if (num1 > = num2) {swap3 (&num1, &num2);}
        printf ("%d, %d", num1, num2);
}
```

A. 5, 5
B. 5, 4
C. 4, 5
D. 4, 4

gate2008-it programming programming-in-c normal parameter-passing

Answer

Answers: Parameter Passing

4.6.1 Parameter Passing: GATE CSE 1992 | Question: 10b top

https://gateoverflow.in/43584

Initially activation stack is empty.

I have used A1, A2, B1, B2 for understanding purpose, which refers to A and B respectively.

```
Test()
{//Scope of Test begins. In activation record Test is added
    A()
    {
    //Scope of A begins.
    // Activation stack: Test---> A
        B1()
        {
        //Scope of B1 begins.
        // Activation stack: Test---> A--->B1
           A1()
            {
           //Scope of A1 begins
            // Activation stack: Test---> A--->B1--->A1
            }//Scope of A1 ends
            //Activation record before y pt of execution Test---> A--->B1

At y point of execution
            A2()
            {
            //New Activation record of A created
            //Activation stack: Test---> A--->B1--->A2
            }//Scope of A2 ends
        }//End of scope B

//Activation stack: Test---> A
            B2()
            {
            //Activation of B2 added
            //Activation stack: Test---> A--->B2
            }
        }//End of scope A
        //Activation Record before x point of execution: Test

At x point of execution
    A()
    {
    //Activation Record: Test--->A
    }
}//End of scope Test
```

0 votes -- SatyamK (225 points)

4.6.2 Parameter Passing: GATE CSE 1994 | Question: 1.20 top https://gateoverflow.in/305

✔ Correct Option: **C**

Passing an array element as a parameter is the answer.

Consider this function call

```
{
    ....
    a[] = {0,1,2,3,4};
    i = 0;
    fun(a[i]);
    print a[0];
}

fun(int x)
{
    int i = 1;
    x = 8;
}
```

Output:

- call-by-reference: 8
- call-by-name: 0

In Call-by-name, each occurrence of the formal parameter is replaced by the actual argument text. So, the function fun will be executed like:

```
{
    int i = 1;
    a[i] = 8; //a[1] is changed to 8 and not a[0]
}
```

A very good read: http://courses.cs.washington.edu/courses/cse341/03wi/imperative/parameters.html

34 votes -- Arjun Suresh (330k points)

4.6.3 Parameter Passing: GATE CSE 2001 | Question: 2.17 | UGCNET-AUG2016-III: 21 top

✔ Answer is B.

Here, variable x of func1 points to address of variable y.

and variables y and z of func1 points to address of variable x.

Therefore, $y = y + 4 \implies y = 10 + 4 = 14$

and $z = x + y + z \implies z = 14 + 14 + 3 = 31$

z will be stored back in x. Hence, $x = 31$ and y will remain as it is. $(y = 3)$

Answer is $31, 3$

27 votes -- jayendra (6.7k points)

4.6.4 Parameter Passing: GATE CSE 2003 | Question: 73 top https://gateoverflow.in/960

✔ **Answer : D**

First refer the following question on Call-by-name parameter passing technique then solve this question.
https://gateoverflow.in/43575/gate2003-74?show=338119#a338119

Call by Name vs. Call by Need :

Assume X is the formal and e the corresponding actual expression.

Call-by-Name :

1. Delays evaluation of arguments past call until a reference to the formal.
2. **Re-evaluates argument e** on each reference to X **in environment of caller.**

3. No local variable X is allocated

Call-by-Need :

1. Delays evaluation of arguments past call until a reference to the formal.

2. Evaluates e **on 1st reference in environment of caller** & loads local variable X; **no re-evaluation:** subsequent references use local X

Since "Call by need" parameter passing technique, it is almost same as Call-by-name But the difference is that Actual argument is evaluated only once(on the first reference) and then that value is saved and re-used on further references But the actual argument is Not re-evaluated.

Caller function's Actual argument contains variable i which clashes with called function P's local variable i, hence, we rename called function P's local variable i and change it to i'.

```
global int i=100, j=5;
void P(x) {
    int i'=10;   // this i' refers to the local variable i' in function P.
    print(x+10);  // this is first reference of x, so here, x= i+j, and these i,j refer to i,j in the caller f
    i'=200;    // this i' refers to the local variable i' in function P.
    j=20;      // this j refers to j in the caller function i.e. main function's environment
    print (x);   // this x is second reference, so, we do not replace it with i+j because in call by need, we
}
main() {
    P(i+j);
}
```

In case of Static scoping : 115, 105

In case of Dynamic scoping : 115, 105

Note that there are no local variable i, j in main function, so, when we say that i, j refer to the i, j in main's environment , we mean that If i, j were accessed/updated in main function then depending on the scoping, which i, j would they refer.

Here, in this question, in both static and dynamic scoping case, i, j will refer to the Global variables.

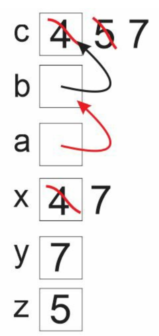
And in function P, in the 4th statement (i.e. $j = 20$), the Global variable j will be updated.

References

18 votes -- Deepak Poonia (23.3k points)

4.6.5 Parameter Passing: GATE CSE 2008 | Question: 60 top

https://gateoverflow.in/483



Return $x + y + z =$ return $7 + 7 + 5 =$ return 19

So, **option B** $= 19$ is correct.

42 votes -- Amar Vashishth (25.2k points)

4.6.6 Parameter Passing: GATE CSE 2010 | Question: 11 top

https://gateoverflow.in/2184

```
  p=q; // now p and q are pointing to same address i.e. address of j
*p=2;// value of j will be updated to 2
```

Hence, answer is (D) 0 2

41 votes -- Manu Thakur (34.1k points)

4.6.7 Parameter Passing: GATE CSE 2013 | Question: 42 top https://gateoverflow.in/60

✔ In GATE 2013 marks were given to all as the same code in C/C++ produces undefined behavior. This is because $*$ is not a sequence point in C/C++. The correct code must replace:

```
return f(x,c) * x;
with
```

```
res = f(x,c); // ';' forms a sequence point
//and all side-effects are guaranteed to be completed here
//-- updation of the x parameter inside f is guaranteed
//to be reflected in the caller from the next point onwards.
return res * x;
```

In this code, there will be 4 recursive calls with parameters $(6,4),(7,3),(8,2)$ and $(9,1)$. The last call returns 1. But due to pass by reference, x in all the previous functions is now 9. Hence, the value returned by $f(p,p)$ will be $9*9*9*9*1=6561$.

Good Read:

- http://stackoverflow.com/questions/41775973/is-this-undefined-behaviour-in-c-if-not-predict-the-output-logically
- https://gateoverflow.in/108445/c-programming?show=108582#a108582

References

78 votes -- Arjun Suresh (330k points)

4.6.8 Parameter Passing: GATE CSE 2016 Set 1 | Question: 15 top https://gateoverflow.in/39642

✔ The mystery about mystery function is it does not affect values in main. As in C, parameters are passed by value- even if they are pointer. So, here the pointer values are exchanged within the function only. (we can use $*$ operator to exchange the values at the location of the pointers and this will affect the values in main).

So, NO CHANGES in a,b,c,d.
And ANSWER is 2016

74 votes -- Abhilash Panicker (7.6k points)

4.6.9 Parameter Passing: GATE CSE 2016 Set 2 | Question: 12 top https://gateoverflow.in/39565

✔ i is called by reference and j is called by value.

So, in function $f()$ only value of i might change,
Now, in function $f(*p,m)$
$*p$ is pointing to i
Thus $*p$ is 5.
m is 10 because of call by value of j.

1. $m=10+5$ hence $m=15$
2. $*p=5+15$ hence $*p=20$, that is, value of variable i is now 20
3. returns nothing

Now, back to main
$i=20$ and j is as it is 10

Hence, output of printf will be $i+j=20+10=30$.

41 votes -- Shashank Chavan (2.4k points)

4.6.10 Parameter Passing: GATE CSE 2018 | Question: 29 top https://gateoverflow.in/204103

✔

```
func1(char* s1, char* s2){
  char* temp;
  temp = s1;
```

```
    s1 = s2;
    s2 = temp;
}
```

Everything is local here. So, once function completes its execution all modification go in vain.

```
func2(char** s1, char** s2){
    char* temp
    temp = *s1
    *s1 = *s2
    *s2 = temp
}
```

This will retain modification and swap pointers of string.
So output would be Hi Bye Bye Hi

Correct Answer: A

32 votes -- Digvijay (44.9k points)

4.6.11 Parameter Passing: GATE IT 2006 | Question: 50 top https://gateoverflow.in/3593

✔ First of all, the swap function just swaps the pointers inside the function and has no effect on the variables being passed.

Inside printab, a and b are added odd integers from 1-5, i.e., $1+3+5=9$. So, in first call to printab, $a=-3+9=6$ and $b=-6+9=3$.

Static variables have one memory throughout program run (initialized during program start) and they keep their values across function calls. So, during second call to printab, $a=6+9=15$, $b=3+9=12$.

Hence, (**D**) is choice.

72 votes -- Arjun Suresh (330k points)

4.6.12 Parameter Passing: GATE IT 2008 | Question: 50 top https://gateoverflow.in/3360

✔ Answer is **C.**

Only:

```
if (num1 > = num2) {swap3 (&num1, &num2);}
```

Statement works, which in turn swaps num1 and num2.

27 votes -- Rajarshi Sarkar (27.8k points)

4.7 Pointers (9) top

4.7.1 Pointers: GATE CSE 2000 | Question: 1.12 top https://gateoverflow.in/635

The most appropriate matching for the following pairs

$X:$	m = malloc(5); m = NULL;	1 :	using dangling pointers
$Y:$	free(n); n -> value = 5;	2 :	using uninitialized pointers
$Z:$	char *p , *p = ‘a’ ;	3 :	lost memory

is:

A. $X-1\ \ Y-3\ \ Z-2$
B. $X-2\ \ Y-1\ \ Z-3$
C. $X-3\ \ Y-2\ \ Z-1$
D. $X-3\ \ Y-1\ \ Z-2$

gate2000-cse programming programming-in-c normal pointers

Answer

4.7.2 Pointers: GATE CSE 2001 | Question: 2.18 top https://gateoverflow.in/736

Consider the following three C functions:

$[P1]$

```
int *g(void)
{
    int x = 10;
    return (&x);
}
```

$[P2]$

```
int *g(void)
{
    int *px;
    *px = 10;
    return px;
}
```

$[P3]$

```
int *g(void)
{
    int *px;
    px = (int*) malloc (sizeof(int));
    *px = 10;
    return px;
}
```

Which of the above three functions are likely to cause problems with pointers?

A. Only $P3$
B. Only $P1$ and $P3$
C. Only $P1$ and $P2$
D. $P1, P2$ and $P3$

gate2001-cse programming programming-in-c normal pointers

Answer

4.7.3 Pointers: GATE CSE 2003 | Question: 2 top

https://gateoverflow.in/893

Assume the following C variable declaration:

```
int *A[10], B[10][10];
```

Of the following expressions:

I. $A[2]$
II. $A[2][3]$
III. $B[1]$
IV. $B[2][3]$

which will not give compile-time errors if used as left hand sides of assignment statements in a C program?

A. I, II, and IV only
B. II, III, and IV only
C. II and IV only
D. IV only

gate2003-cse programming programming-in-c easy pointers

Answer

4.7.4 Pointers: GATE CSE 2003 | Question: 89 top

https://gateoverflow.in/972

Consider the C program shown below:

```
#include<stdio.h>
#define print(x) printf("%d", x)

int x;
void Q(int z)
{
        z+=x;
        print(z);
}
```

```
void P(int *y)
{
        int x = *y + 2;
        Q(x);
        *y = x - 1;
        print(x);
}
main(void) {
        x = 5;
        P(&x);
        print(x);
}
```

The output of this program is:

A. 12 7 6
B. 22 12 11
C. 14 6 6
D. 7 6 6

gate2003-cse programming programming-in-c normal pointers

Answer

4.7.5 Pointers: GATE CSE 2006 | Question: 57 top

https://gateoverflow.in/1835

Consider this C code to swap two integers and these five statements: the code

```
 void swap (int *px, int *py)
    {
       *px = *px - *py;
       *py = *px + *py;
       *px = *py - *px;
    }
```

S1: will generate a compilation error

S2: may generate a segmentation fault at runtime depending on the arguments passed

S3: correctly implements the swap procedure for all input pointers referring to integers stored in memory locations accessible to the process

S4: implements the swap procedure correctly for some but not all valid input pointers

S5: may add or subtract integers and pointers

A. S1
B. S2 and S3
C. S2 and S4
D. S2 and S5

gate2006-cse programming programming-in-c normal pointers

Answer

4.7.6 Pointers: GATE CSE 2014 Set 1 | Question: 10 top

https://gateoverflow.in/1770

Consider the following program in C language:

```
#include <stdio.h>

main()
{
   int i;
   int*pi = &i;

   scanf("%d",pi);
   printf("%d\n", i+5);
}
```

Which one of the following statements is **TRUE?**

A. Compilation fails.
B. Execution results in a run-time error.
C. On execution, the value printed is 5 more than the address of variable i.
D. On execution, the value printed is 5 more than the integer value entered.

gate2014-cse-set1 programming programming-in-c easy pointers

Answer

4.7.7 Pointers: GATE CSE 2015 Set 3 | Question: 26 top https://gateoverflow.in/8478

Consider the following C program

```
#include<stdio.h>
int main()  {
    static int a[] = {10, 20, 30, 40, 50};
    static int *p[] = {a, a+3, a+4, a+1, a+2};
    int **ptr = p;
    ptr++;
    printf("%d%d", ptr-p, **ptr);

}
```

The output of the program is _______.

gate2015-cse-set3 programming programming-in-c normal numerical-answers pointers

Answer

4.7.8 Pointers: GATE CSE 2017 Set 1 | Question: 13 top https://gateoverflow.in/118293

Consider the following C code:

```
#include<stdio.h>
int *assignval (int *x, int val) {
    *x = val;
    return x;
}

void main () {
    int *x = malloc(sizeof(int));
    if (NULL == x) return;
    x = assignval (x,0);
    if (x) {
        x = (int *)malloc(sizeof(int));
        if (NULL == x) return;
        x = assignval (x,10);
    }
    printf("%d\n", *x);
    free(x);
}
```

The code suffers from which one of the following problems:

A. compiler error as the return of $malloc$ is not typecast appropriately.
B. compiler error because the comparison should be made as $x ==$ NULL and not as shown.
C. compiles successfully but execution may result in dangling pointer.
D. compiles successfully but execution may result in memory leak.

gate2017-cse-set1 programming-in-c programming pointers

Answer

4.7.9 Pointers: GATE CSE 2021 Set 2 | Question: 35 top https://gateoverflow.in/357505

Consider the following ANSI C program:

```
#include <stdio.h>
#include <stdlib.h>
struct Node{
        int value;
        struct Node *next;};
int main( ) {
    struct Node *boxE, *head, *boxN; int index=0;
    boxE=head= (struct Node *) malloc(sizeof(struct Node));
    head → value = index;
    for (index =1; index<=3; index++){
        boxN = (struct Node *) malloc (sizeof(struct Node));
        boxE → next = boxN;
        boxN → value = index;
        boxE = boxN; }
```

```
for (index=0; index<=3; index++) {
    printf("Value at index %d is %d\n", index, head → value);
    head = head → next;
    printf("Value at index %d is %d\n", index+1, head → value); } }
```

Which one of the following statements below is correct about the program?

A. Upon execution, the program creates a linked-list of five nodes
B. Upon execution, the program goes into an infinite loop
C. It has a missing `return` which will be reported as an error by the compiler
D. It dereferences an uninitialized pointer that may result in a run-time error

gate2021-cse-set2 programming-in-c normal pointers

Answer

Answers: Pointers

4.7.1 Pointers: GATE CSE 2000 | Question: 1.12 top

https://gateoverflow.in/635

✔ Answer is (**D**).

$X : m = NULL$; makes the pointer m point to $NULL$. But the memory created using $malloc$ is still there and but cannot be used as we don't have a link to it. Hence, lost memory

Y : n is freed and so pointer n is now pointing to an invalid memory making it a Dangling pointer.

Z : p is not initialized. $p = malloc(sizeof(char))$; should have been used before assigning 'a' to $*p$.

72 votes -- Aditi Dan (4k points)

4.7.2 Pointers: GATE CSE 2001 | Question: 2.18 top

https://gateoverflow.in/736

✔ $[P1]$ may cause an error because function is returning the address of locally declared variable.

$[P2]$ will cause a problem because px is an int pointer that is not assigned with any address and we are doing dereferencing.

$[P3]$ will work because memory in bytes of size of int will be reserved and its address will be stored in px that can be further use, once function execution completes, this m/m will still exist in Heap until we free it using $free()$ function.

Hence, answer is (**C**).

83 votes -- Manu Thakur (34.1k points)

4.7.3 Pointers: GATE CSE 2003 | Question: 2 top

https://gateoverflow.in/893

✔ A is an array of pointers to int, and B is a 2-D array.

- $A[2] =$ can take a pointer
- $A[2][3] =$ can take an int
- $B[1] = B[1]$ is the base address of the array and it cannot be changed as the array in C is a constant pointer.
- $B[2][3] =$ can take an integer

So, (**A**) is the answer.

79 votes -- Arjun Suresh (330k points)

4.7.4 Pointers: GATE CSE 2003 | Question: 89 top

https://gateoverflow.in/972

✔

```
main: x = 5; //Global x becomes 5
```

```
P: int x = *y + 2; //local x in P becomes 5+2 = 7
```

```
Q: z+=x; //local z in Q becomes 7 + 5 = 12
```

```
Q: print(z); //prints 12
```

```
P:  *y  = x - 1;
//content of address of local variable y
(same as global variable x) becomes 7 - 1 = 6
```

```
P: print(x); //prints local variable x in P = 7
```

```
main: print(x); //prints the global variable x = 6
```

Correct Answer: A

44 votes -- Arjun Suresh (330k points)

4.7.5 Pointers: GATE CSE 2006 | Question: 57 top https://gateoverflow.in/1835

✔ **S1** is false.

S2 is true, depending on the argument passed it may generate segmentation fault.

S3 is false because implementation is having some problem. Let $x = 3$ and I want to implement $SWAP[x, x]$. Now ans would be 0 but that must be x. Problem is because we are not checking whether both pointer are pointing the same address or different So, S4 is true.

S5 is obviously false so, option (**C**) is right.

73 votes -- Kalpna Bhargav (2.5k points)

4.7.6 Pointers: GATE CSE 2014 Set 1 | Question: 10 top https://gateoverflow.in/1770

✔
```
   int i; //i is declared
int*pi = &i;  //pi is a pointer variable
//and is assigned the address of i

scanf("%d",pi); //i is overwritten with the value
//we provided because pi is pointing to i earlier

printf("%d\n", i+5) //it will print the value stored in i+5
```

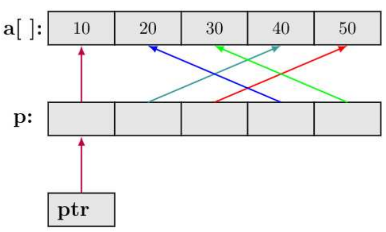
input=3; output=8

Option D is answer.

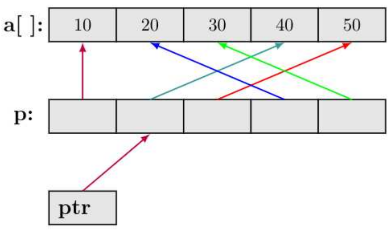
43 votes -- Bhagirathi Nayak (11.7k points)

4.7.7 Pointers: GATE CSE 2015 Set 3 | Question: 26 top https://gateoverflow.in/8478

✔
```
   static int a[] = {10, 20, 30, 40, 50};
static int *p[] = {a, a+3, a+4, a+1, a+2};
int **ptr = p;
```




```
ptr++;
```



$$\text{ptr-p} = \frac{\text{address of ptr} - \text{address of p}}{\text{sizeof(*ptr))}} = 1$$

$$\text{**ptr} = \text{p[2]} = \text{*(a+3)} = 40$$

```
printf("%d%d", ptr-p, **ptr);  // 140
```

👍 49 votes -- Salman (701 points)

4.7.8 Pointers: GATE CSE 2017 Set 1 | Question: 13 top ☛ https://gateoverflow.in/118293

✔ **Answer** is **D.**

Option A: In C++ we need to do typecasting. C does automatic implicit typecasting.

See the screenshot for C & C++ compilers below. C compiler is working fine but C++ compiler is giving error.

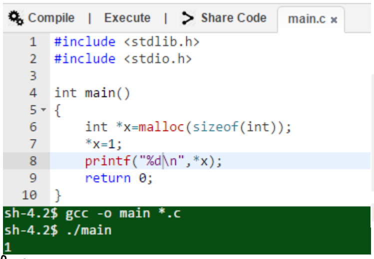
Option B: Null means address 0. if $(a == 0)$ if $(0 == a)$ There is no difference.

Option C: Do it step by step, always x is pointing to a valid memory location. Dangling Pointer means if it points to a memory location which is deleted(freed). So no dangling pointer. http://www.geeksforgeeks.org/dangling-void-null-wild-pointers/

Option D: x will loss the previous address it was pointing to. So it will result in memory leak**.** http://www.geeksforgeeks.org/what-is-memory-leak-how-can-we-avoid/

Proof for **Option A:**

C Compiler:



References

👍 67 votes -- Ahwan Mishra (10.2k points)

4.7.9 Pointers: GATE CSE 2021 Set 2 | Question: 35 top ☛ https://gateoverflow.in/357505

✔ Lets see the first for loop:

```
for (index =1; index<=3; index++){
        boxN = (struct Node *) malloc (sizeof(struct Node));
        boxE -> next = boxN;
        boxN -> value = index;
        boxE = boxN; }
```

After this we get a linked list of size 4 with head pointing to its beginning, and boxE and boxN pointing to the last node and the next pointer of the last node being uninitialized.

Now the second for loop will do printing as follows until the second printf of the final iteration.

- Value at index 0 is 0
- Value at index 1 is 1
- Value at index 1 is 1
- Value at index 2 is 2
- Value at index 2 is 2
- Value at index 3 is 3
- Value at index 3 is 3

After this, the head pointer being uninitialized will be having random content which gets treated as an address. So, when `head -> value` happens it is basically reading data from uninitialized memory location and so can result (not saying **will result** because by chance the uninitialized memory can be a valid location) in runtime error.

To correct the error, we just have to add an extra line of code as given below:

```
for (index =1; index<=3; index++){
```

```
        boxN = (struct Node *) malloc (sizeof(struct Node));
        boxE -> next = boxN;
        boxN -> value= index;
    boxN-> next = NULL;
        boxE = boxN; }
```

Correct option: D

1 votes -- Arjun Suresh (330k points)

4.8 Programming Constructs (1) top

4.8.1 Programming Constructs: GATE CSE 1999 | Question: 2.5 top https://gateoverflow.in/1483

Given the programming constructs

i. assignment
ii. for loops where the loop parameter cannot be changed within the loop
iii. if-then-else
iv. forward go to
v. arbitrary go to
vi. non-recursive procedure call
vii. recursive procedure/function call
viii. repeat loop,

which constructs will you not include in a programming language such that it should be possible to program the terminates (i.e., halting) function in the same programming language

A. (ii), (iii), (iv)
B. (v), (vii), (viii)
C. (vi), (vii), (viii)
D. (iii), (vii), (viii)

gate1999 programming normal programming-constructs

Answer

Answers: Programming Constructs

4.8.1 Programming Constructs: GATE CSE 1999 | Question: 2.5 top https://gateoverflow.in/1483

✔ This question is actually asking about the halting problem of Turing machines. Or in other words which of the constructs are needed to make a programming language Turing complete – when it becomes Turing complete, halting problem becomes undecidable for it.

To start with if we only have a linear sequence of instructions it is guaranteed to terminate because we only have a finite number of instructions to execute and individual instructions can be assumed to finish within a finite time. This is similar to deciding if a TM halts within a finite number of steps or not which is decidable.

The problem (of deciding whether a program halts or not) comes when there is a loop. Again, not all loops are a problem as shown below.

```
int n = 100;
for(int i = 0; i < n; i++)
{
    ...
}
```

Consider the above loop code. We can unroll the loop and repeat the loop body 100 times and what we get is a linear sequence of instructions. So, the above loop does not affect the decision of halting.

Well, in the above paragraph I did not specify one crucial requirement for unrolling the loop. Assume we have a statement like

```
n = pow(n,x;)
```

where x is any program variable. Now, n is changing and so is the bound of the loop and we do not know how many times we have to unroll the loop. (This is similar to a Turing machine tape changing direction from right to left). Does this change make the halting decision undecidable now? “YES” it does. Because now whatever a Turing machine can do we can do in this programming language – Turing complete. So, if we can decide halting problem for this programming language we are indirectly solving the halting problem of Turing machines – which is known to be unsolvable.

So now coming to the given constructs

1. assignment ✓
2. for loops where the loop parameter cannot be changed within the loop ✓
   As described above this just translated to a finite number of sequential instructions when unrolled. Some people might be confused with loops like

```
int n = 0;
for(int i = 1; i > n; )
{

}
```

   Here, if the loop body is not touching either i or n, the loop never terminates. But this decision (that it never terminates) can be decided easily by a written program (analyzing this is decidable and you can think of a C code to do it and equivalently we can have a Turing machine to decide this). So, the above loop even though being non-halting does not make the "halting decision" undecidable.
3. if-then-else ✓
   This just reduces one path (a set of instructions) from the linear sequence of instructions and hence does not affect the halting decision (assuming there are no other structures in either of the paths)
4. forward go to ✓
   Like, if-else this also just eliminates some set of instructions.
5. arbitrary go to ✗
   This can simulate a for loop and so can cause problem in deciding halting.
6. non-recursive procedure call ✓
   Each of the called procedure contributes to the number of executed instructions but since there is no recursion they'll eventually halt as long as each of the called procedures halt.
7. recursive procedure/function call ✗
   This will also run into the same problem of a loop where the loop variables are changed inside the loop body. We may not be able to determine if the sequence of recursive calls ever terminates.
8. repeat loop ✗
   Similar to a for loop if the looping condition is changed within the loop body this can make the halting decision undecidable.

Correct Option: B.

👍 1 votes -- Arjun Suresh (330k points)

4.9 Programming In C (36) top

4.9.1 Programming In C: GATE CSE 2000 | Question: 2.20 top https://gateoverflow.in/667

The value of j at the end of the execution of the following C program:

```
int incr (int i)
{
    static int count = 0;
    count = count + i;
    return (count);
}
main () {
    int i, j;
    for (i = 0; i <= 4; i++)
       j = incr (i);
}
```

is:

A. 10
B. 4
C. 6
D. 7

gate2000-cse programming programming-in-c easy

Answer

4.9.2 Programming In C: GATE CSE 2002 | Question: 1.17 top https://gateoverflow.in/822

In the C language:

A. At most one activation record exists between the current activation record and the activation record for the main
B. The number of activation records between the current activation record and the activation records from the main depends on the actual function calling sequence.
C. The visibility of global variables depends on the actual function calling sequence
D. Recursion requires the activation record for the recursive function to be saved in a different stack before the recursive function can be called.

gate2002-cse programming programming-in-c easy descriptive

Answer

4.9.3 Programming In C: GATE CSE 2002 | Question: 2.18 top

https://gateoverflow.in/848

The C language is:

A. A context free language
B. A context sensitive language
C. A regular language
D. Parsable fully only by a Turing machine

gate2002-cse programming programming-in-c normal

Answer

4.9.4 Programming In C: GATE CSE 2002 | Question: 2.8 top

https://gateoverflow.in/838

Consider the following declaration of a two-dimensional array in C:

char $a[100][100]$;

Assuming that the main memory is byte-addressable and that the array is stored starting from memory address 0, the address of $a[40][50]$ is:

A. 4040
B. 4050
C. 5040
D. 5050

gate2002-cse programming-in-c programming easy

Answer

4.9.5 Programming In C: GATE CSE 2004 | Question: 33 top

https://gateoverflow.in/1030

Consider the following C program segment:

```
char p[20]; int i;
char* s = "string";
int length = strlen(s);
for(i = 0; i < length; i++)
    p[i] = s[length-i];
printf("%s", p);
```

The output of the program is:

A. gnirts
B. string
C. gnirt
D. no output is printed

gate2004-cse programming programming-in-c easy

Answer

4.9.6 Programming In C: GATE CSE 2005 | Question: 1, ISRO2017-55 top

https://gateoverflow.in/1343

What does the following C-statement declare?

```
int (*f) (int * );
```

A. A function that takes an integer pointer as argument and returns an integer
B. A function that takes an integer as argument and returns an integer pointer
C. A pointer to a function that takes an integer pointer as argument and returns an integer
D. A function that takes an integer pointer as argument and returns a function pointer

gate2005-cse programming programming-in-c easy isro2017

Answer

4.9.7 Programming In C: GATE CSE 2005 | Question: 32 top https://gateoverflow.in/1368

Consider the following C program:

```
double foo (double);  /* Line 1 */
int main() {
    double da, db;
    //input da
    db = foo(da);
}
double foo (double a) {
    return a;
}
```

The above code compiled without any error or warning. If Line 1 is deleted, the above code will show:

A. no compile warning or error
B. some compiler-warnings not leading to unintended results
C. some compiler-warnings due to type-mismatch eventually leading to unintended results
D. compiler errors

gate2005-cse programming programming-in-c compiler-design easy

Answer

4.9.8 Programming In C: GATE CSE 2008 | Question: 18 top https://gateoverflow.in/416

Which combination of the integer variables x, y and z makes the variable a get the value 4 in the following expression?

$$a = (x > y)?((x > z)?x : z) : ((y > z)?y : z)$$

A. $x = 3, y = 4, z = 2$
B. $x = 6, y = 5, z = 3$
C. $x = 6, y = 3, z = 5$
D. $x = 5, y = 4, z = 5$

gate2008-cse programming programming-in-c easy

Answer

4.9.9 Programming In C: GATE CSE 2008 | Question: 61 top https://gateoverflow.in/484

Choose the correct option to fill $?1$ and $?2$ so that the program below prints an input string in reverse order. Assume that the input string is terminated by a new line character.

```
void reverse(void)
{
    int c;
    if(?1) reverse();
    ?2
}
main()
{
    printf("Enter text");
    printf("\n");
    reverse();
    printf("\n");
}
```

A. $?1$ is $(getchar()! =' \n')$

?2 is $getchar(c);$

B. ?1 is $((c = getchar())! =' \backslash n')$

?2 is $getchar(c);$

C. ?1 is $(c! =' \backslash n')$

?2 is $putchar(c);$

D. ?1 is $((c = getchar())! =' \backslash n')$

?2 is $putchar(c);$

gate2008-cse programming normal programming-in-c

Answer

4.9.10 Programming In C: GATE CSE 2012 | Question: 3 top

https://gateoverflow.in/35

What will be the output of the following C program segment?

```
char inChar = 'A';
   switch ( inChar ) {
      case 'A' : printf ("Choice A \n");
      case 'B' :
      case 'C' : printf ("Choice B");
      case 'D' :
      case 'E' :
      default : printf ("No Choice");
   }
```

A. No Choice
B. Choice A
C. Choice A
   Choice B No Choice
D. Program gives no output as it is erroneous

gate2012-cse programming easy programming-in-c

Answer

4.9.11 Programming In C: GATE CSE 2012 | Question: 48 top

https://gateoverflow.in/2176

Consider the following C code segment.

```
int a, b, c = 0;
void prtFun(void);
main()
{
    static int a = 1;       /* Line 1 */
    prtFun();
    a += 1;
    prtFun();
    printf(" \n %d %d ", a, b);
}

void prtFun(void)
{
    static int a = 2;       /* Line 2 */
    int b = 1;
    a += ++b;
    printf(" \n %d %d ", a, b);
}
```

What output will be generated by the given code segment?

A.
3 1
4 1
4 2

B.
4 2
6 1
6 1

C.
4 2
6 2
2 0

D.
3 1
5 2
5 2

gate2012-cse programming programming-in-c normal

Answer

4.9.12 Programming In C: GATE CSE 2012 | Question: 49 top https://gateoverflow.in/43314

Consider the following C code segment.

```
int a, b, c = 0;
void prtFun(void);
main()
{
    static int a = 1;       /* Line 1 */
    prtFun();
    a += 1;
    prtFun();
    printf(" \n %d %d ", a, b);
}

void prtFun(void)
{
    static int a = 2;       /* Line 2 */
    int b = 1;
    a += ++b;
    printf(" \n %d %d ", a, b);
}
```

What output will be generated by the given code segment if:

Line 1 is replaced by **auto int** $a = 1$**;**

Line 2 is replaced by **register int** $a = 2$**;**

A.
3 1
4 1
4 2

B.
4 2
6 1
6 1

C.
4 2
6 2
2 0

D.
4 2
4 2
2 0

normal gate2012-cse programming-in-c programming

Answer

4.9.13 Programming In C: GATE CSE 2014 Set 2 | Question: 11 top https://gateoverflow.in/1965

Suppose n and p are unsigned int variables in a C program. We wish to set p to ${}^nC_3$. If n is large, which one of the following statements is most likely to set p correctly?

A. $p = n * (n-1) * (n-2)/6;$
B. $p = n * (n-1)/2 * (n-2)/3;$
C. $p = n * (n-1)/3 * (n-2)/2;$
D. $p = n * (n-1) * (n-2)/6.0;$

gate2014-cse-set2 programming programming-in-c normal

Answer

4.9.14 Programming In C: GATE CSE 2014 Set 2 | Question: 42 top

https://gateoverflow.in/2008

Consider the C function given below.

```
int f(int j)
{
    static int i = 50;
    int k;
    if (i == j)
    {
        printf("something");
        k = f(i);
        return 0;
    }
    else return 0;
}
```

Which one of the following is **TRUE**?

A. The function returns 0 for all values of j.
B. The function prints the string **something** for all values of j.
C. The function returns 0 when $j = 50$.
D. The function will exhaust the runtime stack or run into an infinite loop when $j = 50$.

gate2014-cse-set2 programming programming-in-c

Answer

4.9.15 Programming In C: GATE CSE 2015 Set 1 | Question: 11 top

https://gateoverflow.in/8185

The output of the following C program is_____________.

```
void f1 ( int a, int b)  {
                int c;
                c = a; a = b;
                 b = c;
}
void f2 ( int * a, int * b) {
               int c;
               c = * a; *a = *b; *b = c;
}
int main () {
        int a = 4, b = 5, c = 6;
        f1 ( a, b);
        f2 (&b, &c);
        printf ("%d", c - a - b);
 }
```

gate2015-cse-set1 programming programming-in-c easy numerical-answers

Answer

snore
d

4.9.16 Programming In C: GATE CSE 2015 Set 1 | Question: 35 top

https://gateoverflow.in/8283

What is the output of the following C code? Assume that the address of x is 2000 (in decimal) and an integer requires four bytes of memory.

```
int main () {
         unsigned int x [4] [3] =
         {{1, 2, 3}, {4, 5, 6}, {7, 8, 9}, {10, 11, 12}};
         printf ("%u, %u, %u", x + 3, *(x + 3), *(x + 2) + 3);
}
```

A. $2036, 2036, 2036$
B. $2012, 4, 2204$
C. $2036, 10, 10$
D. $2012, 4, 6$

gate2015-cse-set1 programming programming-in-c normal

Answer

4.9.17 Programming In C: GATE CSE 2015 Set 2 | Question: 15 top https://gateoverflow.in/8086

Consider the following function written in the C programming langauge :

```
void foo(char *a)
 {
    if (*a && *a != ' ')
      {
       foo(a+1);
       putchar(*a);
      }
}
```

The output of the above function on input " $ABCD\ EFGH$ " is

A. $ABCD\ EFGH$
B. $ABCD$
C. $HGFE\ DCBA$
D. $DCBA$

gate2015-cse-set2 programming programming-in-c normal

Answer

4.9.18 Programming In C: GATE CSE 2015 Set 3 | Question: 48 top https://gateoverflow.in/8557

Consider the following C program:

```
#include<stdio.h>
int main()
{
    int i, j, k = 0;
    j=2 * 3 / 4 + 2.0 / 5 + 8 / 5;
    k-=--j;
    for (i=0; i<5; i++)
    {
        switch(i+k)
        {
            case 1:
            case 2: printf("\n%d", i+k);
            case 3: printf("\n%d", i+k);
            default: printf("\n%d", i+k);
        }
    }
    return 0;
}
```

The number of times printf statement is executed is _______.

gate2015-cse-set3 programming programming-in-c normal numerical-answers

Answer

4.9.19 Programming In C: GATE CSE 2015 Set 3 | Question: 54 top https://gateoverflow.in/8563

Consider the following C program:

```
#include<stdio.h>
int f1(void);
int f2(void);
int f3(void);
int x=10;
int main()
{
    int x=1;
    x += f1() + f2 () + f3() + f2();
    printf("%d", x);
    return 0;
}
int f1() { int x = 25; x++; return x;}
int f2() { static int x = 50; x++; return x;}
int f3() { x *= 10; return x;}
```

The output of the program is ______.

gate2015-cse-set3 programming programming-in-c normal numerical-answers

Answer

4.9.20 Programming In C: GATE CSE 2016 Set 1 | Question: 12 top https://gateoverflow.in/39638

Consider the following "C" program.

```
void f(int, short);
void main()
{
    int i = 100;
    short s = 12;
    short *p = &s;
    ____________;  // call to f()
}
```

Which one of the following expressions , when placed in the blank above, will NOT result in a type checking error?

A. $f(s, *s)$
B. $i = f(i, s)$
C. $f(i, *s)$
D. $f(i, *p)$

gate2016-cse-set1 programming-in-c easy

Answer

4.9.21 Programming In C: GATE CSE 2016 Set 1 | Question: 34 top https://gateoverflow.in/39704

The following function computes the maximum value contained in an integer array $P[\,]$ of size n $(n >= 1)$.

```
int max (int *p,int n) {
    int a = 0, b=n-1;

    while (__________) {
        if (p[a]<= p[b]) {a = a+1;}
        else             {b = b-1;}
    }
    return p[a];
}
```

The missing loop condition is:

A. $a\ != n$
B. $b\ != 0$
C. $b > (a + 1)$
D. $b\ != a$

gate2016-cse-set1 programming-in-c normal

Answer

4.9.22 Programming In C: GATE CSE 2017 Set 1 | Question: 53 top https://gateoverflow.in/118473

Consider the following C program.

```
#include<stdio.h>
#include<string.h>

void printlength(char *s, char *t) {
    unsigned int c=0;
    int len = ((strlen(s) - strlen(t)) > c) ? strlen(s) : strlen(t);
    printf("%d\n", len);
}

void main() {
    char *x = "abc";
    char *y = "defgh";
    printlength(x,y);
}
```

Recall that $strlen$ is defined in $string.h$ as returning a value of type $size\_t$, which is an unsigned int. The output of the program is __________ .

gate2017-cse-set1 programming programming-in-c normal numerical-answers

Answer

4.9.23 Programming In C: GATE CSE 2017 Set 1 | Question: 55 top

https://gateoverflow.in/118442

The output of executing the following C program is ________________ .

```
#include<stdio.h>

int total(int v) {
    static int count = 0;
    while(v) {
        count += v&1;
        v >>= 1;
    }
    return count;
}

void main() {
    static int x=0;
    int i=5;
    for(; i>0; i--) {
        x = x + total(i);
    }
    printf("%d\n", x);
}
```

gate2017-cse-set1 programming programming-in-c normal numerical-answers

Answer

4.9.24 Programming In C: GATE CSE 2017 Set 2 | Question: 2 top

https://gateoverflow.in/118171

Match the following:

$P.$	static char var ;	i.	Sequence of memory locations to store addresses
$Q.$	m = malloc(10); m =NULL ;	ii.	A variable located in data section of memory
$R.$	char *ptr[10] ;	iii.	Request to allocate a CPU register to store data
$S.$	register int varl;	iv.	A lost memory which cannot be freed

A. P-ii; Q-iv; R-i; S-iii
B. P-ii; Q-i; R-iv; S-iii
C. P-ii; Q-iv; R-iii; S-i
D. P-iii; Q-iv; R-i; S-ii

gate2017-cse-set2 programming programming-in-c

Answer

4.9.25 Programming In C: GATE CSE 2017 Set 2 | Question: 54 top

https://gateoverflow.in/118272

Consider the following C program.

```
#include<stdio.h>
int main () {
    int m=10;
    int n, n1;
    n=++m;
    n1=m++;
    n--;
    --n1;
    n-=n1;
    printf("%d", n);
    return 0;
}
```

The output of the program is ______

gate2017-cse-set2 programming-in-c numerical-answers

Answer

4.9.26 Programming In C: GATE CSE 2018 | Question: 32 top https://gateoverflow.in/204106

Consider the following C code. Assume that unsigned long int type length is 64 bits.

```
unsigned long int fun(unsigned long int n) {
        unsigned long int i, j=0, sum = 0;
        for( i=n; i>1; i=i/2) j++;
        for( ; j>1; j=j/2) sum++;
        return sum;
}
```

The value returned when we call fun with the input 2^{40} is:

A. 4
B. 5
C. 6
D. 40

gate2018-cse programming-in-c normal programming

Answer

4.9.27 Programming In C: GATE CSE 2019 | Question: 27 top https://gateoverflow.in/302821

Consider the following C program:

```
#include <stdio.h>
int r() {
       static int num=7;
       return num--;
}
int main() {
       for (r();r();r())
              printf("%d",r());
       return 0;
}
```

Which one of the following values will be displayed on execution of the programs?

A. 41
B. 52
C. 63
D. 630

gate2019-cse programming-in-c programming

Answer

4.9.28 Programming In C: GATE CSE 2019 | Question: 52 top https://gateoverflow.in/302796

Consider the following C program:

```
#include <stdio.h>
int main() {
    float sum = 0.0, j=1.0, i=2.0;
    while (i/j > 0.0625) {
        j=j+j;
        sum=sum+i/j;
        printf("%f\n", sum);
    }
    return 0;
}
```

The number of times the variable sum will be printed, when the above program is executed, is ________

gate2019-cse numerical-answers programming-in-c programming

Answer

4.9.29 Programming In C: GATE CSE 2019 | Question: 53 top https://gateoverflow.in/302795

Consider the following C program:

```
#include <stdio.h>
int main()
```

```
{
    int a[] = {2, 4, 6, 8, 10};
    int i, sum=0, *b=a+4;
    for (i=0; i<5; i++)
        sum=sum+(*b-i)-*(b-i);
    printf("%d\n", sum);
    return 0;
}
```

The output of the above C program is _______

gate2019-cse numerical-answers programming-in-c programming

Answer

4.9.30 Programming In C: GATE CSE 2020 | Question: 46 top

☛ https://gateoverflow.in/333185

Consider the following C functions.

`int fun1(int n) {`
`   static int i= 0;`
`   if (n > 0) {`
`      ++i;`
`     fun1(n-1);`
`  }`
`  return (i);`
`}`
	`int fun2(int n) {`
	`   static int i= 0;`
	`   if (n>0) {`
	`      i = i+ fun1 (n) ;`
	`      fun2(n-1) ;`
	`  }`
	`return (i);`
	`}`

The return value of $\text{fun2}(5)$ is _________

gate2020-cse numerical-answers programming-in-c

Answer

4.9.31 Programming In C: GATE CSE 2021 Set 1 | Question: 37 top

☛ https://gateoverflow.in/357414

Consider the following ANSI C program.

```
#include <stdio.h>
int main()
{
    int i, j, count;
    count=0;
    i=0;
    for (j=-3; j<=3; j++)
    {
        if (( j >= 0) && (i++))
            count = count + j;
    }
    count = count +i;
    printf("%d", count);
    return 0;
}
```

Which one of the following options is correct?

A. The program will not compile successfully
B. The program will compile successfully and output 10 when executed
C. The program will compile successfully and output 8 when executed
D. The program will compile successfully and output 13 when executed

gate2021-cse-set1 programming-in-c

Answer

4.9.32 Programming In C: GATE IT 2004 | Question: 59 top

☛ https://gateoverflow.in/3702

What is the output of the following program?

```
#include<stdio.h>
int funcf (int x);
int funcg (int y);
main ()
{
    int x = 5, y = 10, count;
```

```
    for (count = 1; count <= 2; ++count) {
        y += funcf(x) + funcg(x);
        printf ("%d", y);
    }
}
funcf (int x) {
    int y;
    y = funcg(x);
    return (y);
}
funcg (int x) {
    static int y = 10;
    y += 1;
    return (y + x);
}
```

A. 43 80
B. 42 74
C. 33 37
D. 32 32

gate2004-it programming programming-in-c normal

Answer

4.9.33 Programming In C: GATE IT 2004 | Question: 60 top

https://gateoverflow.in/3703

Choose the correct option to fill the ?1 and ?2 so that the program prints an input string in reverse order. Assume that the input string is terminated by a new line character.

```
#include <stdio.h>
void wrt_it (void);
int main (void)
{
    printf("Enter Text");
    printf ("\n");
    wrt_it();
    printf ("\n");
    return 0;
}
void wrt_it (void)
{
    int c;
    if (?1)
        wrt_it();
    ?2
}
```

A. ?1 is $getchar()! =$ '\n'
 ?2 is $getchar(c);$
B. ?1 is $(c = getchar()); ! =$ '\n'
 ?2 is $getchar(c);$
C. ?1 is $c! =$ '\n'
 ?2 is $putchar(c);$
D. ?1 is $(c = getchar())! =$ '\n'
 ?2 is $putchar(c);$

gate2004-it programming programming-in-c normal

Answer

4.9.34 Programming In C: GATE IT 2005 | Question: 58 top

https://gateoverflow.in/3819

Let a be an array containing n integers in increasing order. The following algorithm determines whether there are two distinct numbers in the array whose difference is a specified number $S > 0$.

```
i = 0; j = 1;
while (j < n ){
         if (E) j++;
         else if (a[j] - a[i] == S) break;
         else i++;
}
if (j < n) printf("yes") else printf ("no");
```

Choose the correct expression for E.

A. $a[j] - a[i] > S$
B. $a[j] - a[i] < S$
C. $a[i] - a[j] < S$
D. $a[i] - a[j] > S$

gatee2005-it programming normal programming-in-c

Answer

4.9.35 Programming In C: GATE IT 2006 | Question: 51 top

https://gateoverflow.in/3594

Which one of the choices given below would be printed when the following program is executed?

```
#include <stdio.h>
int a1[] = {6, 7, 8, 18, 34, 67};
int a2[] = {23, 56, 28, 29};
int a3[] = {-12, 27, -31};
int *x[] = {a1, a2, a3};
void print(int *a[])
{
            printf("%d,", a[0][2]);
            printf("%d,", *a[2]);
            printf("%d,", *++a[0]);
            printf("%d,", *(++a)[0]);
            printf("%d\n", a[-1][+1]);
}
main()
{
             print(x);
}
```

A. $8, -12, 7, 23, 8$
B. $8, 8, 7, 23, 7$
C. $-12, -12, 27, -31, 23$
D. $-12, -12, 27, -31, 56$

gate2006-it programming programming-in-c normal

Answer

4.9.36 Programming In C: GATE IT 2007 | Question: 31 top

https://gateoverflow.in/3464

Consider the C program given below :

```
#include <stdio.h>
int main ()    {
    int sum = 0, maxsum = 0,  i,  n = 6;
    int a [] = {2, -2, -1, 3, 4, 2};
    for (i = 0; i < n; i++)    {
            if (i == 0 || a [i]  < 0  || a [i] < a [i - 1])  {
                     if (sum > maxsum) maxsum = sum;
                     sum = (a [i] > 0) ? a [i] : 0;
            }
            else sum += a [i];
    }
    if (sum > maxsum) maxsum = sum ;
    printf ("%d\n", maxsum);

}
```

What is the value printed out when this program is executed?

A. 9
B. 8
C. 7
D. 6

gate2007-it programming programming-in-c normal

Answer

Answers: Programming In C

4.9.1 Programming In C: GATE CSE 2000 | Question: 2.20 top https://gateoverflow.in/667

✔ Answer: (**A**)

- At $i = 0, j = 0$
- At $i = 1, j = 1$
- At $i = 2, j = 3$
- At $i = 3, j = 6$
- At $i = 4, j = 10$

29 votes -- Rajarshi Sarkar (27.8k points)

4.9.2 Programming In C: GATE CSE 2002 | Question: 1.17 top https://gateoverflow.in/822

✔

A. Each function call starts a new activation record and since C allows nested function calls more than one activation record can exist between the current activation record and main.
B. **TRUE**
C. Since, C uses static scoping, the actual function calling sequence has no impact on the visibility of global variables. If a variable is not found in the current activation record, it is looked in global address space and this is independent of the calling sequence.
D. All function calls- whether recursive or not uses the same stack for saving the activation record. There is no need for a different stack as for C compiler a recursive function call and a normal function call make no difference.

73 votes -- Arjun Suresh (330k points)

4.9.3 Programming In C: GATE CSE 2002 | Question: 2.18 top https://gateoverflow.in/848

✔ Answer is (**B**).

All modern programming languages are CSL. Because they contain two features which cannot be handled by PDA.

The features are:

- variable declared before use and
- matching formal and actual parameters of functions.

59 votes -- Sachin Mittal (15.8k points)

4.9.4 Programming In C: GATE CSE 2002 | Question: 2.8 top https://gateoverflow.in/838

✔ The answer is (**B**).

In C, arrays are always stored in the row-major form.

Formula to evaluate 2-D array's location is:

$$loc(a[i][j]) = BA + [(i - lb_1) \times NC + (j - lb_2)] \times c$$

Where,

- BA - Base Address
- NC - no. of columns
- c - memory size allocated to data type of array $a[lb_1 \dots ub_1][lb_2 \dots ub_2]$

Here, $\text{BA} = 0, \text{NC} = 100, c = 1, a[0 \dots 99][0 \dots 99]$, so $lb_1 = 0, lb_2 = 0$

$$\begin{aligned} loc(a[40][50]) &= 0 + [(40 - 0) \times 100 + (50 - 0)] \times 1 \\ &= 0 + [4000 + 50] \times 1 = 4050. \end{aligned}$$

41 votes -- Kalpna Bhargav (2.5k points)

4.9.5 Programming In C: GATE CSE 2004 | Question: 33 top https://gateoverflow.in/1030

Here,

$p[0] = s[length] =$ '\0'; //compiler puts a '\0' at the end of all string literals

Now, for any string function in C, it checks till the first '\0' to identify the end of string. So, since the first char is '\0', printf %s, will print empty string. If we use printf("%s", p+1); we will get option (C) with some possible garbage until some memory location happens to contain "\0". For the given code, answer is (**D**).

40 votes -- Arjun Suresh (330k points)

4.9.6 Programming In C: GATE CSE 2005 | Question: 1, ISRO2017-55 top https://gateoverflow.in/1343

A. A function that takes an integer pointer as argument and returns an integer $\Rightarrow int\ f(int*)$
B. A function that takes an integer as argument and returns an integer pointer $\Rightarrow int * f(int)$
C. A pointer to a function that takes an integer pointer as argument and returns an integer $\Rightarrow$

```
int (*f) (int * );
```

So, answer is **C.**

43 votes -- Akash Kanase (36k points)

4.9.7 Programming In C: GATE CSE 2005 | Question: 32 top https://gateoverflow.in/1368

Answer is (**D**).

When a function is called without being defined, C compiler assumes it to return "int" but here foo is returning "$double$" and hence the compiler will throw type mis-match error.

From $C99$ on ward, implicit declaration of functions is not even allowed.

59 votes -- Aditi Dan (4k points)

4.9.8 Programming In C: GATE CSE 2008 | Question: 18 top https://gateoverflow.in/416

Using option (A) : $x = 3, y = 4, z = 2$
$a = (3 > 4)?$ No
therefore don't evaluate the first part and check second part $((y > z)?y : z)$
$(4 > 2)?$ Yes
$a =$ value of $y = 4$
Answer is (A) $x = 3, y = 4, z = 2$

26 votes -- Keith Kr (4.5k points)

4.9.9 Programming In C: GATE CSE 2008 | Question: 61 top https://gateoverflow.in/484

Here, we are using the '$=$' operator which has less priority than '$!=$' operator. So $(c = getchar())$ has to be in brackets and after reversing the string we use function $putchar(c)$ for printing the character.

So, option (**D**) is the right answer.

27 votes -- Kalpna Bhargav (2.5k points)

4.9.10 Programming In C: GATE CSE 2012 | Question: 3 top https://gateoverflow.in/35

There is a \`space\` in between the \`\\\` and \`n\`. (see-Q-no.-3)

```
case 'A' : printf ("Choice A\ n");                    ^
```

So, output of the given program is:

```
Choice A nChoice BNo Choice
```

Which includes:

```
'n'
```

And there is no new line or spaces between outputs. Hence, there is no option matching.

http://stackoverflow.com/questions/33694700/im-missing-something

References

33 votes -- Arjun Suresh (330k points)

4.9.11 Programming In C: GATE CSE 2012 | Question: 48 top https://gateoverflow.in/2176

✔ main
$a = 1$
$prtFun()$
$a = 2$
$b = 1$
$a = a + ++b = 2 + 2 = 4$
$b = 2$
printf $\rightarrow$ 4 2
back to main
$a = a + 1 \rightarrow 1 + 1 \rightarrow 2$ (local static a is taken)
$prtFun()$
$a = 4$ // previous value in the function is retained because of static
$b = 1$
$a = a + ++b = 4 + 2 = 6$
$b = 2$
printf $\rightarrow$ 6 2
back to main
$a = 2$
$b = 0$ (initial value of global b. in $prtFun$ local b is only updated)
printf $\rightarrow$ 2 0

The answer is **C.**

56 votes -- Sankaranarayanan P.N (8.5k points)

4.9.12 Programming In C: GATE CSE 2012 | Question: 49 top https://gateoverflow.in/43314

✔ main
$a = 1$

$prtFun()$
$a = 2$
$b = 1$
$a = a + ++b = 2 + 2 = 4$
$b = 2$
printf $\rightarrow$ 4 2
back to main
$a = a + 1 \rightarrow 1 + 1 \rightarrow 2$

$prtFun()$
$a = 2$ //previous a is lost
$b = 1$
$a = a + ++b = 2 + 2 = 4$
$b = 2$
printf $\rightarrow$ 4 2

back to main

$a = 2b = 0$ (initial value of global b. in $prtFun$ local b is only updated)
printf $\rightarrow$ 2 0

Answer is **D.**

29 votes -- Sankaranarayanan P.N (8.5k points)

4.9.13 Programming In C: GATE CSE 2014 Set 2 | Question: 11 top https://gateoverflow.in/1965

Answer is (B).
In c, $*$ and $/$ have the same precedence and are left associative.

Evaluation of $n * (n-1) * (n-2)$ might exceed the unsigned int range.
So, **(A)** and **(D)** are eliminated.
$n * (n-1)$ is always divisible by 2.(Gives an integer value). Where as it is not always divisible by 3.(You don't always get an integer..truncation possible, less accuracy)
(C) eliminated.

In option **(B)**
$n * (n-1)/2$ gives an integer value.
This integer value multiplied by $(n-2)$ again gives an integer value.
Which when divided by 3 gives an integer value(Sets p correctly).

Reason : $n * (n-1) * (n-2)$ is the multiplication of 3 consecutive numbers. which is divisible by 2 as well as 3.
Hence, $(n * (n-1)/2 * (n-2))$ is divisible by 3.

193 votes -- Srinath Jayachandran (2.9k points)

4.9.14 Programming In C: GATE CSE 2014 Set 2 | Question: 42 top https://gateoverflow.in/2008

There is no updation for i and j in the function. so if we call function with $j = 50$ the recursive call will be continued infinitely. There is no terminating condition for recursion. hence answer **D**

38 votes -- Sankaranarayanan P.N (8.5k points)

4.9.15 Programming In C: GATE CSE 2015 Set 1 | Question: 11 top https://gateoverflow.in/8185

```
void f1 ( int a, int b)  {     //This code is call by value
// hence no effect of actual values when run.
                int c;
                c = a;
                a = b;
                b = c;
}
void f2 ( int * a, int * b) {    //*a= address of b
//and *b = address of c
               int c;              //int c = garbage
               c = * a;            //c = value at address a = 5;
               *a = *b;            //*a = Exchange original
// variable value of c to b = b= 6
               *b = c;               //*b = c = 5;
}
int main () {
        int a = 4, b = 5, c = 6;
        f1 ( a, b);  This has no effect on actual values
// of a ,b since Call by value.
        f2 (&b, &c); Here change will be happen.
        At this point  int a = 4, b = 6, c = 5;
        printf ("%d", c - a - b);     = (5-4)-6 = 1-6 = -5
 }
```

14 votes -- Anu007 (14.4k points)

Here, $f1$ will not change any values bcz it is call by value but $f2$ is call by reference and it swaps values of b and c and changes are also reflected in main function. So, $5 - 4 - 6 = -5$ is the answer.

34 votes -- target gate (111 points)

4.9.16 Programming In C: GATE CSE 2015 Set 1 | Question: 35 top https://gateoverflow.in/8283

✔ Address of x is 2000.

x being a $2\ D$ array,

$x+3 = x+3*$ sizeof its inner dimension

$= 2000+3*3*4$ (as inner dimension is 3 integers of size 4)

$= 2000+36 = 2036.$

$*(x+3)$ returns the value at address 2036. But since x is $2-D$ array, one $*$ will just return the $1\ D$ array which is the starting address of it, which is 2036 only.

$(x+2) = 2000+2*3*4 = 2024$
$*(x+2)+3 = 2024+3*4 = 2036$ (The $*$ changes the data type from $2D$ to $1D$ and hence $+3$ will add $3*4$ and not $3*3*4$)

So, **A**.

138 votes -- Arjun Suresh (330k points)

4.9.17 Programming In C: GATE CSE 2015 Set 2 | Question: 15 top https://gateoverflow.in/8086

✔ Answer **D** as priority of $!=$ is greater than that of $\&\&$ in **C**. The execution happens as:

```
if ((*a) && (*a != ' '))
```

So, the if breaks either when $*a = 0$ (not '0' but ASCII 0 or null character '\0'), or when $*a = ' '$.

So, the recursive call goes like

'A' - 'B' - 'C' - 'D' -' ' (breaks) and then starts outputting

$DCBA$

46 votes -- Vikrant Singh (11.2k points)

4.9.18 Programming In C: GATE CSE 2015 Set 3 | Question: 48 top https://gateoverflow.in/8557

✔
```
j=2 * 3 / 4 + 2.0 / 5 + 8 / 5;
```

$j = (((2*3)/4)+(2.0/5))+(8/5)$; //As associativity of $+, *$ and $/$ are from left to right and $+$ has less precedence than $*$ and $/$.

$j = ((6/4)+0.4)+1)$; //2.0 is double value and hence 5 is implicitly typecast to double and we get 0.4. But 8 and 5 are integers and hence $8/5$ gives 1 and not 1.6

$j = (1+0.4)+1$; // $6/4$ also gives 1 as both are integers

$j = 1.4+1$; //$1+0.4$ gives 1.4 as 1 will be implicitly typecast to 1.4

$j = 2.4$; // since j is integer when we assign 2.4 to it, it will be implicitly typecast to int.

So, $j = 2$;

$k-=--j$;

This makes $j=1$ and $k=-1$.

The variables j and k have values 1 and -1 respectively before the for loop. Inside the for loop, the variable i is initialized to 0 and the loop runs from 0 to 4.

$i=0, k=-1, i+k=-1$, default case is executed, printf $count = 1$

$i=1, k=-1, i+k=0$, default case is executed, printf $count = 2$

$i=2, k=-1, i+k=1$, case 2, case 3 and default case is executed, printf $count = 5$

$i=3, k=-1, i+k=2$, case 2, case 3 and default case is executed, printf $count = 8$

$i=4, k=-1, i+k=3$, case 3 and default case is executed, printf $count = 10$

$i=5$, loop exits and the control returns to main

Answer is **10**.

106 votes -- Shyam Singh (1.3k points)

4.9.19 Programming In C: GATE CSE 2015 Set 3 | Question: 54 top https://gateoverflow.in/8563

✔ The variable x is initialized to 1. First and only call to $f1()$ returns 26. First call to $f2()$ returns 51. First and only call to $f3()$ returns 100. Second call to $f2()$ returns 52 (The value of local static variable x in $f2()$ retains its previous value 51 and is incremented by 1).

$$x = 1 + 26 + 51 + 100 + 52 = 230$$

Answer: 230

52 votes -- Shyam Singh (1.3k points)

4.9.20 Programming In C: GATE CSE 2016 Set 1 | Question: 12 top https://gateoverflow.in/39638

✔

A. $f(s, *s)$ - 1st argument is short whereas the function expects an int. But in C language short gets implicitly converted to int during a function call and so this won't be a type error. But second argument is a pointer (can be 64 bits on a x64 machine) where as the function expects a short (can be even 16 bits). So this will generate a type error. So, WRONG.
B. $i = f(i, s)$ - return type is not void. So, WRONG.
C. $f(i, *s)$ - 1st argument is int, second is again syntax error. So, WRONG
D. $f(i, *p)$ - Both the arguments and return type match. p is a pointer to short, so $*p$ is value of short. So, **D** is **ANSWER.**

70 votes -- Abhilash Panicker (7.6k points)

4.9.21 Programming In C: GATE CSE 2016 Set 1 | Question: 34 top https://gateoverflow.in/39704

✔ Answer is (**D**).

Hint : Given in the question itself that we start comparing the contents of an array from $a[0]$ and $a[n-1]$ (converging from both side) then condition must be till both meet at a point and that point will be $a = b$.
Hence loop condition should be $a! = b$.

Option C fails for $n = 2, p = [1, 2]$.

45 votes -- sukanyac (131 points)

4.9.22 Programming In C: GATE CSE 2017 Set 1 | Question: 53 top https://gateoverflow.in/118473

✔ $(strlen(s) - strlen(t)) = 3 - 5 = -2$

But in C, when we do operation with two unsigned integers, result is also unsigned. ($strlen$ returns $size\_t$ which is unsigned in most systems). So, this result "-2" is treated as unsigned and its value is $INT\_MAX - 2$ (not sure about all systems, but at least on systems using 2's complement representation). Now, the comparison is between this large number and another unsigned number c which is 0. So, the comparison return TRUE here.

Even if 'c' is declared as "int", while doing an operation with an unsigned int, it gets promoted to unsigned int and we get the same result.

Hence $(strlen(s) - strlen(t)) > 0$ will return 1 on execution thus the conditional operator will return the true statement which is $strlen(abc) = 3$.

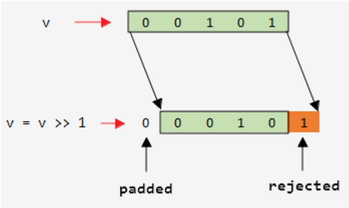
Ans should be 3.

99 votes -- Arnabi Bej (5.8k points)

4.9.23 Programming In C: GATE CSE 2017 Set 1 | Question: 55 top https://gateoverflow.in/118442

✔

```
   // inside total()

while(v) {
        count += v&1;   \\ check the lowest bit of v
        v >>= 1;        \\ or v = v >> 1 : right shift the bit pattern of v
}
```



This piece of code will count the no of set bits in v.

In the `main` function, `i` values goes from 5 to 1, So there will be 5 calls to `total()`.

Each call to `total(i)` counts the no of set bits in `i`. But the **count** is a `static` variable,

So, `total(i)` $=$ Total no of set bits in all $i \leq 5$.

$$\mathrm{x} = \sum_{i=1}^{5} [\,\mathrm{total}\,(i)] = 2 + 3 + 5 + 6 + 7 = \mathbf{23}$$

51 votes -- Debashish Deka (40.7k points)

4.9.24 Programming In C: GATE CSE 2017 Set 2 | Question: 2 top

https://gateoverflow.in/118171

✔ static char var $=$ A variable **located in data section** of memory

$m = malloc(10); m = null;$ **Here, free(m) is missing:** So, a lost memory which cannot be freed

$char * Ptr[10];$ Sequence of **10 memory locations** to store addresses

register int var1; Request to allocate a **CPU register** to store data

Answer is **A.**

27 votes -- Prashant Singh (47.1k points)

4.9.25 Programming In C: GATE CSE 2017 Set 2 | Question: 54 top

https://gateoverflow.in/118272

✔

- $m = 10;$ $\quad \overset{m}{\boxed{10}}$
- $n = ++m;$ $\quad \overset{n}{\boxed{11}} \; \overset{m}{\boxed{11}}$
- $n1 = m++;$ $\quad \overset{n1}{\boxed{11}} \; \overset{m}{\boxed{12}}$
- n --; $\quad \overset{n}{\boxed{10}}$
- -- $n1$; $\quad \overset{n1}{\boxed{10}}$
- n -= $n1$; $\quad \overset{n}{\boxed{0}} \quad (\because n = n - n1 = 10 - 10 = 0)$

So,

```
printf("%d"", n);
```

prints "0".

Hence, answer is **0.**

👍 29 votes -- Arnabi Bej (5.8k points)

4.9.26 Programming In C: GATE CSE 2018 | Question: 32 top

https://gateoverflow.in/204106

✔

```
unsigned long int fun(unsigned long int n) {
        unsigned long int i, j=0, sum = 0;
        for( i=n; i>1; i=i/2) j++;
        for( ; j>1; j=j/2) sum++;
        return sum;
}
```

First for loop will make $j = 40$

Next for loop will divide j value (which is 40 now) by 2 each time until $j \leq 1$.

```
j loop starts:
j = 40 and sum =1,
j = 20 and sum=2,
j = 10 and sum=3,
j = 5  and sum=4,
j = 2  and sum=5,
j=1 break
```

So, Sum$=5$

Correct Answer: B

👍 34 votes -- Digvijay (44.9k points)

4.9.27 Programming In C: GATE CSE 2019 | Question: 27 top

https://gateoverflow.in/302821

✔ Basic Points :-

1. after every expression statement in the for loop there is a sequence point
2. After return value copied, there is a sequence point.

for loop execution :- **for(e1;e2;e3)**

on first iteration, expression1 executed (generally termed as initialization expression.)

next expression2 executed (generally termed as condition check expression, if it evaluate to non-zero then only, for loop body executed, otherwise for loop terminates.)

after first iteration, expression3 executed (generally termed as increment expression.), after that e2 evaluates and process continues !

for$(r();r();r())$

{

printf("%d",r());

}

before main starts the execution **num** initialized with 7 (note that it is stored under static memory due to it is static number.)

on first iteration :- $r() \implies$ return 7 but **num** changed to 6.

$r() \implies$ return 6 but **num** changed to 5. $\implies$ condition evaluate to true $\implies$ for loop body executes !

printf("%d",r()); $\implies$ return 5 but **num** changed to 4. $\implies$ print 5 on the screen.

$r() \implies$ return 4 but **num** changed to 3.

$r() \implies$ return 3 but **num** changed to 2. $\implies$ condition evaluate to true $\implies$ for loop body executes !

printf("%d",r()); $\implies$ return 2 but **num** changed to 1. $\implies$ print 2 on the screen.

$r() \implies$ return 1 but **num** changed to 0.

$r() \implies$ return 0 but **num** changed to -1. $\implies$ condition evaluate to false $\implies$ for loop over !

Hence option B : 52

👍 60 votes -- Shaik Masthan (50.4k points)

4.9.28 Programming In C: GATE CSE 2019 | Question: 52 top https://gateoverflow.in/302796

✔ $i = 2.0, j = 1.0$

while $(\frac{i}{j} > 0.0625)$

$j = 1$
$\frac{i}{j} = \frac{2}{1} > 0.0625$
$j = j + j, 1^{st}$ PRINT

$j = 2$
$\frac{i}{j} = \frac{2}{2} > 0.0625$
$j = j + j, 2^{nd}$ PRINT

$j = 4$
$\frac{i}{j} = \frac{2}{4} > 0.0625$
$j = j + j, 3^{rd}$ PRINT

$j = 8$
$\frac{i}{j} = \frac{2}{8} > 0.0625$
$j = j + j, 4^{th}$ PRINT

$j = 16$
$\frac{i}{j} = \frac{2}{16} > 0.0625$
$j = j + j, 5^{th}$ PRINT

$j = 32$
$\frac{i}{j} = \frac{2}{32} = 0.0625$
$Break$

Total 5 times sum will be printed.

25 votes -- Digvijay (44.9k points)

4.9.29 Programming In C: GATE CSE 2019 | Question: 53 top https://gateoverflow.in/302795

✔ sum $= 0, *b = a + 4$ i.e.pointing to 10

sum $=$ sum $+ (*b - i) - *(b - i)$

$i = 0$
sum $= 0 + (10 - 0) - (10) = 0$

$i = 1$
sum $= 0 + (10 - 1) - (8) = 1$

$i = 2$
sum $= 1 + (10 - 2) - (6) = 3$

$i = 3$
sum $= 3 + (10 - 3) - (4) = 6$

$i = 4$
sum $= 6 + (10 - 4) - (2) = 10$

23 votes -- Digvijay (44.9k points)

4.9.30 Programming In C: GATE CSE 2020 | Question: 46 top https://gateoverflow.in/333185

✔ This illustration of function calling and values may help😀

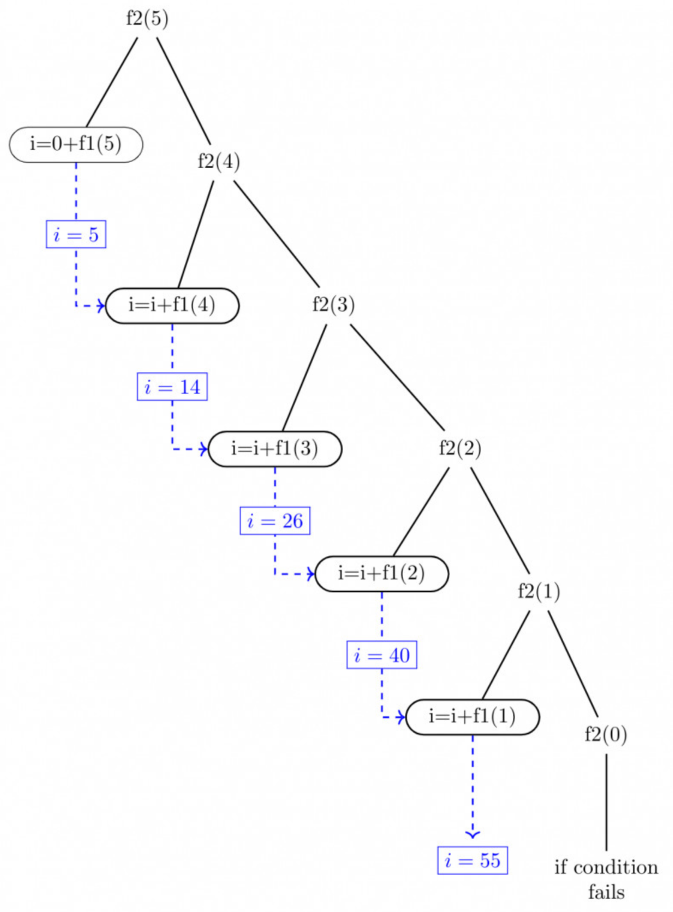


Fig : Tracing by tree method

Now $f1(5)$, if we observe it increments 'i', n times so $f1(5)$ will return 5.

Similarly, $f1(4)$ will increment 'i' 4 times but 'i' being static (just one memory location for entire program run instead of different memory location across function calls), it'll resume from its previous value of 5. So, $f1(4)$ returns 9 $[5+4]$

By the same logic $f1(3)$ will return 12 $[9+3]$,

$f1(2)$ will return 14 $[12+2]$,

$f1(1)$ will return 15 $[14+1]$.

$\therefore$ Return value $i = 55$ and hence 55 is the output of $f2(5)$.

10 votes -- Asim Siddiqui (2.2k points)

4.9.31 Programming In C: GATE CSE 2021 Set 1 | Question: 37 top

https://gateoverflow.in/357414

```
for (j=-3; j<=3; j++)
{
    if (( j >= 0) && (i++))
        count = count + j;
}
```

From the above loop code we can see that the loop iterates 7 times for $j \in \{-3, -2, -1, 0, 1, 2, 3\}$.

Now, we have an "if" condition and inside it we have a logical AND operator (&&). In C language we have the following short-circuit rule for binary logical operators

1. The second operand of logical OR operator `||` is ignored if the first operand is non zero.
2. The second operand of logical AND operator `(&&)` is ignored if the first operand is 0.

So, for $j \in \{-3, -2, -1\}$ the first operand of `||` operator `(j >= 0)` will be 0, and hence the second operand `(i++)` will be ignored.

For $j \in \{0, 1, 2, 3\}$ the first operand of `||` operator `(j >= 0)` will be 1, and hence the second operand `(i++)` will get evaluated 4 times and final value of $i = 4$.

Initial value of $i = 0$.

The postincrement operator `i++`, returns the original value of i and then increments i. So, when the first time `i++` happens, the second operator of logical AND operator is 0 and hence the "if" condition fails. So, `count = count +j` happens only for $j \in \{1, 2, 3\}$ and we get $\text{count} = 0 + 1 + 2 + 3 = 6$.

After the loop, we have `count = count + i`, which makes $\text{count} = 6 + 4 = 10$.

So, correct option: B.

Reference: https://gateoverflow.in/62409/what-is-the-output

References

1 votes -- gatecse (62.6k points)

4.9.32 Programming In C: GATE IT 2004 | Question: 59 top

https://gateoverflow.in/3702

```
funcf(x) + funcg(x)
```

$funcf$ or $funcg$ can be executed first as whether the first operand or the second operand to '+' operator is first executed is not defined in C language and it depends on the compiler implementation. But here the order does not matter as both the operands are not affecting each other and '+' is commutative. Lets assume $funcf$ is executed first. It calls $funcg$ - so even if the order of call is reversed, result will be same.

In first call of $funcg$, y becomes 11 and it returns $5 + 11 = 16$.

In second call of $funcg$, y becomes 12 and it returns $5 + 12 = 17$.

So, in main y is incremented by $16 + 17 = 33$ to become $10 + 33 = 43$. (Choice **A**)

In the second iteration y will be incremented by $18 + 19 = 37$ to give $43 + 37 = 80$.

40 votes -- Arjun Suresh (330k points)

4.9.33 Programming In C: GATE IT 2004 | Question: 60 top

https://gateoverflow.in/3703

$getchar()$ - reads a single character at a time from the $stdin$.

$putchar(c)$ - writes a character specified by the argument to $stdout$.

As
$getchar()$ and
$putchar()$ both are needed to read the string and print its reverse and only option D contains both the function. D is the answer. :P

Now coming to the code.

$wrt\_it(void)$ is calling itself recursively. when \n is encountered $putchar()$ gets executed and prints the last character and then the function returns to its previous call and prints last 2nd character and so on.

19 votes -- Soumya Jain (12.5k points)

4.9.34 Programming In C: GATE IT 2005 | Question: 58 top

https://gateoverflow.in/3819

Answer is (B)

For some 'i' if we find that difference of $(A[j] - A[i] < S)$ we increment 'j' to make this difference wider so that it becomes equal to S .

If at times difference becomes greater than S we know that it wont reduce further for same 'i' and so we increment the 'i'.

We do it for each 'i' if not found in previous iteration. until $i = n$

43 votes -- Sandeep_Uniyal (6.5k points)

4.9.35 Programming In C: GATE IT 2006 | Question: 51 top

https://gateoverflow.in/3594

$a = \{a1, a2, a3\}$;

```
printf("%d,", a[0][2]);
```

$a[0]$ is $a1$. So, this will print $a1[2] = 8$;

```
printf("%d,", *a[2]);
```

$a[2]$ is $a3$. So, this will print $*a3 = a3[0] = -12$ ([] has greater precedence than $*$)

```
printf("%d,", *++a[0]);
```

$a[0]$ which is $a1$ is incremented. $a1$ is a pointer to int (base address of an integer array) and so increment means adding $sizeof(int)$ and hence $a1$ now points to the second element in the array. So, $*++a[0]$ prints second element of $a1$ which is 7 and now $a1$ starts from 7.

```
printf("%d,", *(++a)[0]);
```

$++a$ will increment a, which being a pointer (In C, an array when passed to a function becomes a pointer) to pointer (to int) will add $sizeof(pointer)$ to a. So, a now contains $\{a2, a3\}$ and $a[0]$ will be $a2$ and $*a2$ will be the first element in $a2$ which is 23

```
printf("%d\n", a[-1][+1]);
```

$a[-1]$ will subtract a size of pointer from the base address of a. Normally this results in invalid memory access, but since we have incremented a previously, $a[-1]$ is valid and will point to $a1$. So, $a[-1][+1]$ will be $a1[1]$ which has the value 8. ($a1$ was incremented in 3rd printf and hence starts from 7 and not 6. $+1$ is same as 1, just given to create confusion)

Correct Answer: A

108 votes -- Arjun Suresh (330k points)

4.9.36 Programming In C: GATE IT 2007 | Question: 31 top

https://gateoverflow.in/3464

✔ Answer is **C.**

i	$A[i]$	for $\rightarrow$ if --- satisfied?	maxsum	sum
-	-	-	0	0
0	$A[0] = 2$	Yes $(i == 0)$	0	2
1	$A[1] = -2$	Yes $(a[i] < 0)$	2	0
2	$A[2] = -1$	Yes $(a[i] < 0)$	(for $\rightarrow$ if $\rightarrow$ if --- not satisfied)	0
3	$A[3] = 3$	No (else executed)	(for $\rightarrow$ if $\rightarrow$ if --- not satisfied)	3
4	$A[4] = 4$	No (else executed)	(for $\rightarrow$ if $\rightarrow$ if --- not satisfied)	7
5	$A[5] = 2$	Yes $(a[i] < a[i-1])$	7	2

[End of for loop]

If (sum (i.e., 2) $>$ maxsum (i.e., 7)) // No

maxsum $=$ sum; // Not Executed

printf will output maxsum $= 7$

31 votes -- MKT (141 points)

The algorithm is finding the maximum sum of the monotonically increasing continuous sequence of positive numbers in the array. So, output would be $3 + 4 = 7$.

24 votes -- Arjun Suresh (330k points)

4.10 Programming Paradigms (2) top

4.10.1 Programming Paradigms: GATE CSE 2004 | Question: 1 top

https://gateoverflow.in/998

The goal of structured programming is to:

A. have well indented programs
B. be able to infer the flow of control from the compiled code
C. be able to infer the flow of control from the program text
D. avoid the use of GOTO statements

gate2004-cse programming easy programming-paradigms

Answer

4.10.2 Programming Paradigms: GATE CSE 2004 | Question: 90 top

https://gateoverflow.in/1084

Choose the best matching between the programming styles in Group 1 and their characteristics in Group 2.

Group 1		Group 2	
P.	Functional	1.	Common-based, procedural
Q.	Logic	2.	Imperative, abstract data types
R.	Object-oriented	3.	Side-effect free, declarative, expression evaluations
S.	Imperative	4.	Declarative, clausal representation, theorem proving

A. $P-2 \quad Q-3 \quad R-4 \quad S-1$
B. $P-4 \quad Q-3 \quad R-2 \quad S-1$
C. $P-3 \quad Q-4 \quad R-1 \quad S-2$
D. $P-3 \quad Q-4 \quad R-2 \quad S-1$

gate2004-cse programming normal programming-paradigms

Answer

Answers: Programming Paradigms

4.10.1 Programming Paradigms: GATE CSE 2004 | Question: 1 top https://gateoverflow.in/998

✔ Answer is (**C**). The goal of structured programming is to able to infer the flow of control from the program text . It means user can execute the code according to his requirement. C and Pascal are good example of structured programming. In structured programming control passes one instruction to another instruction in sequential manner.

Avoiding the use of GOTO statements is not the goal of structured programming, it (avoiding the use of GOTO) is one of the requirements for a program to be structured.

35 votes -- Kalpna Bhargav (2.5k points)

4.10.2 Programming Paradigms: GATE CSE 2004 | Question: 90 top https://gateoverflow.in/1084

✔ Answer: (D) P-3 Q-4 R-2 S-1

Group 1		Group 2	
P.	Functional	3.	Side-effect free, declarative, expression evaluations
Q.	Logic	4.	Declarative, clausal representation, theorem proving
R.	Object-oriented	2.	Imperative, abstract data types
S.	Imperative	1.	Common-based, procedural

Explanation:

P: Functional Programming is declarative in nature, involves expression evaluation, & side effect free.
Q: Logic is also declarative but involves theorem proving.
R: Object-oriented is an imperative statement based & have abstract (general) data types.
S: Imperative The programs are made giving commands & follows definite procedure & sequence

Ref: https://www.geeksforgeeks.org/gate-gate-cs-2004-question-90/

References

19 votes -- Siddharth Mahapatra (1.2k points)

4.11 Recursion (15) top

4.11.1 Recursion: GATE CSE 1991 | Question: 01,x top https://gateoverflow.in/507

Consider the following recursive definition of fib:

```
fib(n) := if n = 0 then 1
          else if n = 1 then 1
          else fib(n-1) + fib(n-2)
```

The number of times fib is called (including the first call) for evaluation of $fib(7)$ is___________.

gate1991 programming recursion normal numerical-answers

Answer

4.11.2 Recursion: GATE CSE 1994 | Question: 21 top

☛ https://gateoverflow.in/2517

Consider the following recursive function:

```
function fib (n:integer);integer;
begin
if (n=0) or (n=1) then fib := 1
else fib := fib(n-1) + fib(n-2)
end;
```

The above function is run on a computer with a stack of 64 bytes. Assuming that only return address and parameter are passed on the stack, and that an integer value and an address takes 2 bytes each, estimate the maximum value of n for which the stack will not overflow. Give reasons for your answer.

gate1994 programming recursion normal descriptive

Answer

4.11.3 Recursion: GATE CSE 2000 | Question: 16 top

☛ https://gateoverflow.in/687

A recursive program to compute Fibonacci numbers is shown below. Assume you are also given an array $f[0\dots m]$ with all elements initialized to 0.

```
fib(n) {
    if (n > M) error ();
    if (n == 0) return 1;
    if (n == 1)return 1;
    if (▭)________________________(1)
        return ▭__________________(2)
    t = fib(n - 1) + fib(n - 2);
    ▭_____________________________(3)
    return t;
}
```

A. Fill in the boxes with expressions/statement to make $fib()$ store and reuse computed Fibonacci values. Write the box number and the corresponding contents in your answer book.
B. What is the time complexity of the resulting program when computing $fib(n)$?

gate2000-cse algorithms normal descriptive recursion

Answer

4.11.4 Recursion: GATE CSE 2001 | Question: 13 top

☛ https://gateoverflow.in/754

Consider the following C program:

```
void abc(char*s)
{
    if(s[0]=='\0')return;
    abc(s+1);
    abc(s+1);
    printf("%c",s[0]);
}

main()
{
    abc("123");
}
```

A. What will be the output of the program?
B. If $abc(s)$ is called with a null-terminated string s of length n characters (not counting the null ('\0') character), how many characters will be printed by $abc(s)$?

gate2001-cse programming recursion normal descriptive

Answer

4.11.5 Recursion: GATE CSE 2002 | Question: 11 top https://gateoverflow.in/864

The following recursive function in C is a solution to the Towers of Hanoi problem.

```
void move(int n, char A, char B, char C)  {
    if (......................) {
        move (............................);
        printf("Move disk %d from pole %c to pole %c\n", n, A, C);
        move (.....................);
    }
}
```

Fill in the dotted parts of the solution.

gate2002-cse programming recursion descriptive

Answer

4.11.6 Recursion: GATE CSE 2004 | Question: 31, ISRO2008-40 top https://gateoverflow.in/1028

Consider the following C function:

```
int f(int n)
{
    static int i = 1;
    if(n >= 5) return n;
    n = n+i;
    i++;
    return f(n);
}
```

The value returned by $f(1)$ is:

A. 5
B. 6
C. 7
D. 8

gate2004-cse programming programming-in-c recursion easy isro2008

Answer

4.11.7 Recursion: GATE CSE 2005 | Question: 81b top https://gateoverflow.in/82146

```
double foo(int n)
{
    int i;
    double sum;
    if(n == 0)
    {
        return 1.0;
    }
    else
    {
        sum = 0.0;
        for(i = 0; i < n; i++)
        {
            sum += foo(i);
        }
        return sum;
    }

}
```

Suppose we modify the above function $foo()$ and stores the value of $foo(i)$ $0 \leq i < n$, as and when they are computed. With this modification the time complexity for function $foo()$ is significantly reduced. The space complexity of the modified function would be:

A. $O(1)$
B. $O(n)$
C. $O(n^2)$
D. $n!$

gate2005-cse programming recursion normal

Answer

4.11.8 Recursion: GATE CSE 2007 | Question: 42 top

https://gateoverflow.in/1240

Consider the following C function:

```
int f(int n)
{
    static int r = 0;
    if (n <= 0) return 1;
    if (n > 3)
    {   r = n;
        return f(n-2) + 2;
    }
    return f(n-1) + r;
}
```

What is the value of $f(5)$?

A. 5
B. 7
C. 9
D. 18

gate2007-cse programming recursion normal

Answer

4.11.9 Recursion: GATE CSE 2014 Set 2 | Question: 40 top

https://gateoverflow.in/2000

Consider the following function.

```
double f(double x){
    if( abs(x*x - 3) < 0.01)
        return x;
    else
        return f(x/2 + 1.5/x);
}
```

Give a value q (to 2 decimals) such that $f(q)$ will return q:_____.

gate2014-cse-set2 programming recursion numerical-answers normal

Answer

4.11.10 Recursion: GATE CSE 2016 Set 1 | Question: 35 top

https://gateoverflow.in/39730

What will be the output of the following C program?

```
void count (int n) {
    static int d=1;

    printf ("%d",n);
    printf ("%d",d);
    d++;
    if (n>1) count (n-1);
    printf ("%d",d);


}

void main(){
    count (3);
}
```

Variables like 'd' are always computed at last

A. 3 1 2 2 1 3 4 4 4
B. 3 1 2 1 1 1 2 2 2
C. 3 1 2 2 1 3 4
D. 3 1 2 1 1 1 2

gate2016-cse-set1 programming-in-c recursion normal

Answer

4.11.11 Recursion: GATE CSE 2016 Set 2 | Question: 37 top https://gateoverflow.in/39602

Consider the following program:

```
int f (int * p, int n)
{
    if (n <= 1) return 0;
    else return max (f (p+1, n-1), p[0] - p[1]);
}
 int main ()
{
    int a[] = {3, 5, 2, 6, 4};
    print f(" %d", f(a, 5));
}
```

Note: $max(x, y)$ returns the maximum of x and y.

The value printed by this program is ________.

gate2016-cse-set2 programming-in-c normal numerical-answers recursion

Answer

4.11.12 Recursion: GATE CSE 2017 Set 1 | Question: 35 top https://gateoverflow.in/118317

Consider the following two functions.

```
void fun1(int n) {
    if(n == 0) return;
    printf("%d", n);
    fun2(n - 2);
    printf("%d", n);
}
void fun2(int n) {
    if(n == 0) return;
    printf("%d", n);
    fun1(++n);
    printf("%d", n);
}
```

The output printed when $\text{fun1}(5)$ is called is

A. 53423122233445
B. 53423120112233
C. 53423122132435
D. 53423120213243

gate2017-cse-set1 programming normal tricky recursion

Answer

4.11.13 Recursion: GATE CSE 2017 Set 1 | Question: 36 top https://gateoverflow.in/118319

Consider the C functions foo and bar given below:

```
int foo(int val) {
    int x=0;
    while(val > 0) {
        x = x + foo(val--);
    }
    return val;
}
```

```
int bar(int val) {
    int x = 0;
    while(val > 0) {
        x= x + bar(val-1);
    }
    return val;
}
```

Invocations of $foo(3)$ and $bar(3)$ will result in:

A. Return of 6 and 6 respectively.
B. Infinite loop and abnormal termination respectively.
C. Abnormal termination and infinite loop respectively.

D. Both terminating abnormally.

gate2017-cse-set1 programming-in-c programming normal recursion

Answer

4.11.14 Recursion: GATE CSE 2018 | Question: 21 top https://gateoverflow.in/204095

Consider the following C program:

```
#include<stdio.h>

int counter=0;

int calc (int a, int b) {
        int c;
        counter++;
        if(b==3) return (a*a*a);
        else {
                c = calc(a, b/3);
                return (c*c*c);
        }
}

int main() {
        calc(4, 81);
        printf("%d", counter);
}
```

The output of this program is ______.

gate2018-cse programming-in-c numerical-answers recursion programming

Answer

4.11.15 Recursion: GATE IT 2007 | Question: 27 top https://gateoverflow.in/3460

The function f is defined as follows:

```
int f (int n) {
    if (n <= 1) return 1;
    else if (n % 2  ==  0) return f(n/2);
    else return f(3n - 1);
}
```

Assuming that arbitrarily large integers can be passed as a parameter to the function, consider the following statements.

i. The function f terminates for finitely many different values of $n \geq 1$.
ii. The function f terminates for infinitely many different values of $n \geq 1$.
iii. The function f does not terminate for finitely many different values of $n \geq 1$.
iv. The function f does not terminate for infinitely many different values of $n \geq 1$.

Which one of the following options is true of the above?

A. i and iii
B. i and iv
C. ii and iii
D. ii and iv

gate2007-it programming recursion normal

Answer

Answers: Recursion

4.11.1 Recursion: GATE CSE 1991 | Question: 01,x top https://gateoverflow.in/507

✔ The recurrence relation for the number of calls is

$$T(n) = T(n-1) + T(n-2) + 1$$

where 1 is for the current called function.

- $T(0) = T(1) = 1$ (for $fib(0)$ and $fib(1)$, there are no recursive calls and only current function call is there).
- $T(2) = 3$
- $T(3) = 5$
- $T(4) = 9$
- $T(5) = 15$
- $T(6) = 25$
- $T(7) = 41$

Answer: 41

48 votes -- Arjun Suresh (330k points)

4.11.2 Recursion: GATE CSE 1994 | Question: 21 top https://gateoverflow.in/2517

✔ Size of an activation record $= 2 + 2 = 4$ bytes.

So, no. of possible activation records which can be live at a time $= 64/4 = 16$.

So, we can have 16 function calls live at a time (recursion depth $= 16$), meaning the maximum value for n without stack overflow is 16 (calls from $1 - 16$). For $n = 17$, stack will overflow.

This is different from the total no. of recursive calls which will be as follows:

n	No. of calls
1	1
2	3
3	5
4	9
5	15
6	25

50 votes -- Arjun Suresh (330k points)

4.11.3 Recursion: GATE CSE 2000 | Question: 16 top https://gateoverflow.in/687

✔ Array f is used to store the $fib()$ values calculated in order to save repeated calls. Since $n = 0$ and $n = 1$ are special cases we can store $fib(2)$ to $f[0]$, $fib(3)$ to $f[1]$ and so on. The missing code completed would be:

```
if (f[n - 2] != 0){
    return f[n-2];
}
t = fib(n-1) + fib(n-2);
f[n-2] = t;
return t;
```

In this code, $fib(i)$ will do a recursion only once as once $fib(i)$ is calculated it is stored in array. So, the time complexity for $fib(n)$ would be $\Theta(n)$.

PS: We can also store $fib(n)$ in $f(n)$, the above code just saves 2 elements' space in the array.

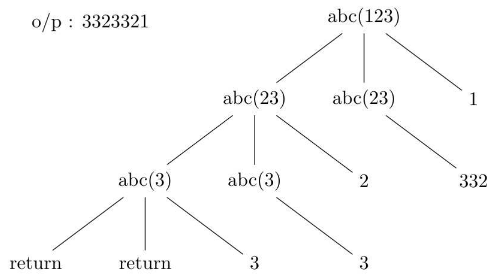
33 votes -- Arjun Suresh (330k points)

4.11.4 Recursion: GATE CSE 2001 | Question: 13 top https://gateoverflow.in/754

✔ Answer (A) : 332 332 1
Answer (B) : $2^n - 1$

Q. (A) O/p :

3323321



(B) : $T(n) = 2T(n-1) + 1; n > 0$

$= 0; n = 0$ [Since for length zero string no character will be printed]

After solving it by substitution,

$$T(n) = 2T(n-1) + 1$$
$$= 2(2T(n-2) + 1) + 1$$
$$= 2^2T(n-2) + 2 + 1$$
$$= 2^2(2T(n-3) + 1) + 2 + 1$$
$$= 2^3T(n-3) + 2^2 + 2 + 1$$

Finally, it will expand like

$$T(n) = 2^nT(n-n) + 2^{n-1} + 2^{n-2} + \ldots + 2^2 + 2 + 1$$
$$= 2^nT(0) + 2^{n-1} + 2^{n-2} + \ldots + 2^2 + 2 + 1$$
$$= \frac{1.(2^n-1)}{(2-1)}$$
$$= 2^n - 1$$

34 votes -- jayendra (6.7k points)

4.11.5 Recursion: GATE CSE 2002 | Question: 11 top https://gateoverflow.in/864

```
  void move(int n, char A, char B, char C) {
   if (n > 0) {
       move (n-1, A, C, B);
       printf("Move disk %d from pole %c to pole %c\n", n, A, C);
       move (n-1, B, A, C);
   }
}
```

25 votes -- minal (13.1k points)

4.11.6 Recursion: GATE CSE 2004 | Question: 31, ISRO2008-40 top https://gateoverflow.in/1028

Answer is 7. As,

$f(1) : n = 2, i = 2$

$f(2) : n = 4, i = 3$

$f(4) : n = 7, i = 4$

$f(7) : print(n) \Rightarrow 7$ <ans>

34 votes -- sumit kumar singh dixit (1.6k points)

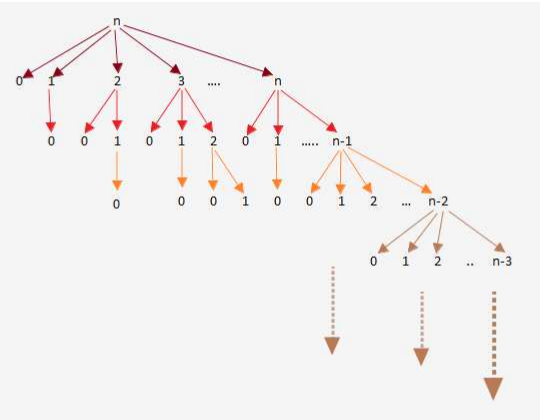
4.11.7 Recursion: GATE CSE 2005 | Question: 81b top https://gateoverflow.in/82146

Given program:

```
#include <stdio.h>
double foo(int n) {
 int i;
 double sum;
 if(n == 0) {
     return 1.0;
```

```
 }
 else {
  sum = 0.0;
  for(i = 0; i < n; i++) {
      sum += foo(i);
  }
  return sum;
 }
}

int main() {
 double a = foo(5);
 printf("%.2f\n",a);
}
```

And here is the present situation :



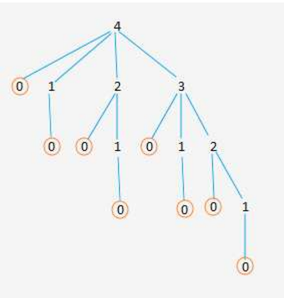
Here we can see that, we have lots of overlapping subproblems. Too many function calls.

1. No of function calls = 2^n
2. stack depth used = $O(n)$

Therefore space is linear and time is exponential.

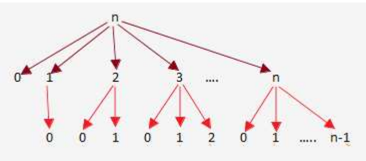
If we take a small number say 4, then we would have 8.0 as answer, or we can see that $foo(n) = 2^{n-1}$ and

1. stack depth used = 5.
2. No of function calls = $2^4 = 16$.



Now, using one-dimensional $(1D)$ table we can reduce the no of function calls and depth of stack space used as well.

Here is what we want to achieve:



We are reusing already computed $foo(x)$ values. For this purpose, we will be using one $1D$ array of doubles of size n.

Here is what we are going to do:

1. First, check in the $1D$ table for the required call value.
2. If correct value found: do not call recursive functions

3. If not, then only attempt `for loop` recursive calls

Here is the code:

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <math.h>
#define UNUSED 0.5
double *memo_table;
double foo(int n) {
 int i;
 double sum;
 if(memo_table[n] != UNUSED) {
  return memo_table[n];
 }else {
  sum = 0.0;
  for(i=0;i<n;i++) {
   sum += foo(i);
  }
  return memo_table[n] = sum;
 }
}
int main() {
 int n,i;
 scanf("%d",&n);
 memo_table = malloc((1+n)*sizeof(double));
 for(i=0;i<=n;i++) memo_table[i] = UNUSED;
 // base case
 memo_table[0] = 1.0;
 double a = foo(n);
 printf("%lf\n",a);
 free(memo_table);
}
```

Improvements over given program:

1. Stack space because of function calls reduced to two level only.
2. Extra space used for the $1D$ array = $O(n)$
3. More importantly, Time is reduced to $O(n^2)$. (Exponential to Quadratic !!)

Overall space complexity = stack + extra = $O(1) + O(n) = O(n)$

Answer is **B.**

49 votes -- Debashish Deka (40.7k points)

4.11.8 Recursion: GATE CSE 2007 | Question: 42 top

https://gateoverflow.in/1240

✔ The answer is **D**.

$f(5) = 18.$

$f(3) + 2 = 16 + 2 = 18$

$f(2) + 5 = 11 + 5 = 16$

$f(1) + 5 = 6 + 5 = 11$

$f(0) + 5 = 1 + 5 = 6$

Consider from last to first. Since it is recursive function.

31 votes -- Gate Keeda (15.9k points)

4.11.9 Recursion: GATE CSE 2014 Set 2 | Question: 40 top

https://gateoverflow.in/2000

✔ (We can directly go to the "if" part to get one answer, but we need to solve "else" part too to get all possible answers which though is not asked in question)

Solving the else part:

$\frac{x}{2} + \frac{3}{2x} = \frac{x^2+3}{2x}$

So, the new value of x will be $\frac{x^2+3}{2x}$ and we need it equal to x.

$\frac{x^2+3}{2x} = x \implies x^2 + 3 = 2x^2 \implies x^2 = 3 \implies x = 1.732$

Now solving the if part.

```
abs(x*x - 3) < 0.01
```

So, $x^2 - 3 < 0.01$ and $-\left(x^2 - 3\right) < 0.01 \implies x^2 < 3.01$ and $x^2 > 2.99 \implies x < 1.735$ and $x > 1.729$

Corrected to 2 decimal places answer should be 1.73 or 1.74.

78 votes

-- Arjun Suresh (330k points)

4.11.10 Recursion: GATE CSE 2016 Set 1 | Question: 35 top

https://gateoverflow.in/39730

✔ Here, **d** is **Static**, so the value of **d** will persists between the function calls.

1. $count(3)$ will print the value of n and d and increments d and call $count(2)$ $\Rightarrow prints$ 3 1.
2. $count(2)$ will print the value of n and d and increments d and call $count(1)$ $\Rightarrow prints$ 2 2.
3. $count(1)$ will print the value of n and d and increments $d \Rightarrow$ $prints$ 1 3.

Now, it will return and prints the final incremented value of d which is 4, three times.
So, option **(A)** is correct = 3 1 2 2 1 3 4 4 4

54 votes

-- Monanshi Jain (7k points)

4.11.11 Recursion: GATE CSE 2016 Set 2 | Question: 37 top

https://gateoverflow.in/39602

✔ $f(a, 5)$

- p , $n = 5$. $\boxed{3}\boxed{5}\boxed{2}\boxed{6}\boxed{4}$
  $\max(f(p + 1, 5 - 1), 3 - 5) = \max(f(p + 1, 4), -2)$
- p , $n = 4$. $\boxed{5}\boxed{2}\boxed{6}\boxed{4}$
  $\max(\max(f(p + 1, 4 - 1), 5 - 2), -2) = \max(\max(f(p + 1, 3), 3), -2)$
- p , $n = 3$. $\boxed{2}\boxed{6}\boxed{4}$
  $\max(\max(\max(f(p + 1, 3 - 1), 2 - 6), 3), -2) = \max(\max(\max(f(p + 1, 2), -4), 3), -2)$
- p , $n = 2$. $\boxed{6}\boxed{4}$
  $\max(\max(\max(\max(f(p + 1), 1), 2), -4), 3), -2)$
- $n = 1$, return 0

$\max(\max(\max(\max(0, 2), -4), 3), -2)$

$= \max(\max(\max(2, -4), 3), -2)$

$= \max(\max(2, 3), -2)$

$= \max(3, -2)$

$= 3$

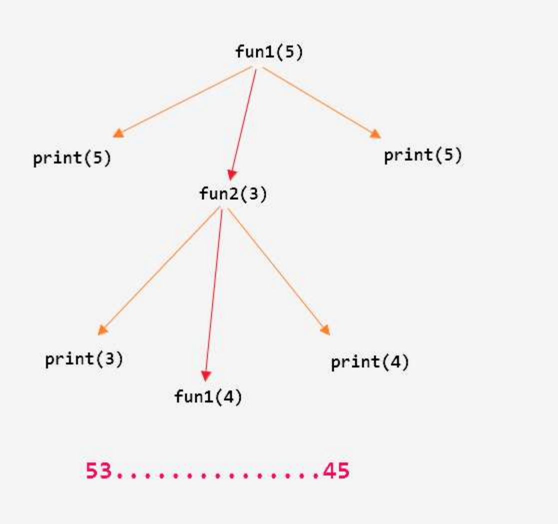
55 votes

-- Praveen Saini (41.9k points)

4.11.12 Recursion: GATE CSE 2017 Set 1 | Question: 35 top

https://gateoverflow.in/118317

✔



- Unroll recursion up to a point where we can distinguish the given options and choose the correct one!
- Options **B** and **D** are eliminated.
- **A** is the answer.

77 votes

-- Debashish Deka (40.7k points)

4.11.13 Recursion: GATE CSE 2017 Set 1 | Question: 36 top

https://gateoverflow.in/118319

✔ **Answer should be C.**

Consider foo

Initially $val = 3$

$foo(val--)$

is equivalent to

1. $foo(val)$
2. $val = val - 1$

So,

1. $foo(3)$
2. $val = 2$

$foo(3)$ calls $foo(3)$ which in turn calls $foo(3)$ this goes on

So, here we can see that $foo(3)$ is called infinite number of times which causes memory overflow and abrupt termination and one more thing to observe is infinite loop is not there since the val is decremented in the first iteration.

Consider bar.

Here, we can see the val is not decrementing in the loop

So,

1. $bar(3)$ will call $bar(2)$
2. $bar(2)$ will call $bar(1)$
3. $bar(1)$ will call $bar(0)$ $\quad \rightarrow$ Here, $bar(0)$ $return$ 0
4. $bar(1)$ will call $bar(0)$
5. $bar(1)$ will call $bar(0)$

This will go on so here there is a problem of infinite loop but not abrupt termination since it does not cause memory overflow.

81 votes

-- (points)

4.11.14 Recursion: GATE CSE 2018 | Question: 21 top https://gateoverflow.in/204095

```
int main() {
        calc(4, 81);
        printf("%d", counter);
}
```

```
printf("%d", counter);
So we need only counter value.

Each function increments counter value by 1. Goal is to find the number of function cal
Squence of function calls:
calc(4, 81) ---> calc(4, 27) ---> calc(4, 9) ---> calc(4, 3) ---> return

4 function calls.

counter = 4
```

34 votes -- Digvijay (44.9k points)

4.11.15 Recursion: GATE IT 2007 | Question: 27 top https://gateoverflow.in/3460

The function terminates for all powers of 2 (which is infinite), hence (i) is false and (ii) is TRUE.

Let $n = 5$.
Now, recursive calls will go like $5 - 14 - 7 - 20 - 10 - 5 -$

And this goes into infinite recursion. And if we multiply 5 with any power of 2, also result will be infinite recursion. Since, there are infinite powers of 2 possible, there are infinite recursions possible (even considering this case only). So, (iv) is TRUE and (iii) is false.

So, correct answer is (**D**).

55 votes -- Arjun Suresh (330k points)

4.12 Structures (4) top

4.12.1 Structures: GATE CSE 2000 | Question: 1.11 top https://gateoverflow.in/634

The following C declarations:

```
struct node {
    int i:
    float j;
 };
 struct node *s[10];
```

define s to be:

A. An array, each element of which is a pointer to a structure of type node
B. A structure of 2 fields, each field being a pointer to an array of 10 elements
C. A structure of 3 fields: an integer, a float, and an array of 10 elements
D. An array, each element of which is a structure of type node

gate2000-cse programming programming-in-c easy structures

Answer

4.12.2 Structures: GATE CSE 2018 | Question: 2 top https://gateoverflow.in/204076

Consider the following C program:

```
#include<stdio.h>
struct Ournode{
    char x, y, z;
};
int main() {
    struct Ournode p={'1', '0', 'a'+2};
    struct Ournode *q=&p;
    printf("%c, %c", *((char*)q+1), *((char*)q+2));
    return 0;
}
```

The output of this program is:

A. 0, c
B. 0, a+2
C. '0', 'a+2'
D. '0', 'c'

gate2018-cse programming-in-c programming structures pointers normal

Answer

4.12.3 Structures: GATE IT 2004 | Question: 61 top

https://gateoverflow.in/3704

Consider the following C program:

```
#include <stdio.h>
typedef struct {
    char *a;
    char *b;
    } t;
void f1 (t s);
void f2 (t *p);
main()
{
    static t s = {"A", "B"};
    printf ("%s %s\n", s.a, s.b);
    f1(s);
    printf ("%s %s\n", s.a, s.b);
    f2(&s);
}
void f1 (t s)
{
    s.a = "U";
    s.b = "V";
    printf ("%s %s\n", s.a, s.b);
    return;
}
void f2(t *p)
{
    p -> a  = "V";
    p -> b = "W";
    printf("%s %s\n", p -> a, p -> b);
    return;
}
```

What is the output generated by the program ?

A. $A\ B$
$U\ V$
$V\ W$
$V\ W$
B. $A\ B$
$U\ V$
$A\ B$
$V\ W$
C. $A\ B$
$U\ V$
$U\ V$
$V\ W$
D. $A\ B$
$U\ V$
$V\ W$
$U\ V$

gate2004-it programming programming-in-c normal structures

Answer

4.12.4 Structures: GATE IT 2006 | Question: 49 top

https://gateoverflow.in/3592

Which one of the choices given below would be printed when the following program is executed ?

```
#include <stdio.h>
struct test {
            int i;
```

THE BEST ANSWER IS LEGENDARY.MOST UPVOTED ONE

```
        char *c;
}st[] = {5, "become", 4, "better", 6, "jungle", 8, "ancestor", 7, "brother"};
main ()
{
    struct test *p = st;
    p += 1;
    ++p -> c;
    printf("%s,", p++ -> c);
    printf("%c,", *++p -> c);
    printf("%d,", p[0].i);
    printf("%s \n", p -> c);
}
```

A. jungle, n, 8, nclastor
B. etter, u, 6, ungle
C. cetter, k, 6, jungle
D. etter, u, 8, ncestor

gate2006-it programming programming-in-c normal structures

Answer

Answers: Structures

4.12.1 Structures: GATE CSE 2000 | Question: 1.11 top

https://gateoverflow.in/634

✔ Correct Option: **A**

[] has greater precedence than $*$ in C. So, s becomes an array of pointers.

49 votes -- gatecse (62.6k points)

4.12.2 Structures: GATE CSE 2018 | Question: 2 top

https://gateoverflow.in/204076

✔ char $x =$'a' $+2$
i.e. $x ==$'c'

So, $p =${'1','0','c'}

$*((char*)q+1) == p[1]$
$*((char*)q+2) == p[2]$

```
printf("%c, %c", *((char*)q+1), *((char*)q+2));
```

Output: $0, c$

Correct Answer: A

38 votes -- Digvijay (44.9k points)

4.12.3 Structures: GATE IT 2004 | Question: 61 top

https://gateoverflow.in/3704

✔ First print $A\ B$

$f1$ is call by value the changes applciable only for local

from $f1\ U\ V$ is printed

back in main $A\ B$ is printed

then in $f2\ V\ W$ is printed

Hence, answer is (**B**).

33 votes -- Sankaranarayanan P.N (8.5k points)

4.12.4 Structures: GATE IT 2006 | Question: 49 top

https://gateoverflow.in/3592

✔ code :

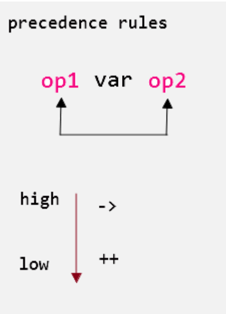
```c
#include <stdio.h>

struct test {
    int i;
    char *c;
}st[] = {5, "become", 4, "better", 6, "jungle", 8, "ancestor", 7, "brother"};

int main () {
    //printf("size = %d\n",sizeof(struct test) );
    struct test *p = st;
    p += 1;
    ++p->c; // ++(p->c)
    printf("%s,", p++->c); // (p++)->c
    printf("%c,", *++p->c); // *(++(p->c))
    printf("%d,", p[0].i);
    printf("%s \n", p->c);
}
```

We will assume few things:

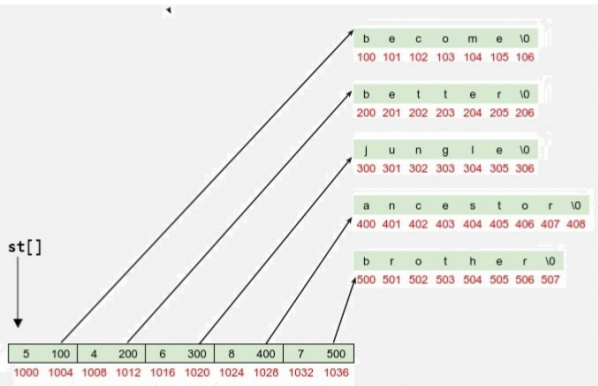
- Size of integer 4 Bytes.
- Size of a pointer 4 Bytes.

Neglecting any alignment issues with the storage of this structure we will have 8 Bytes per structure.

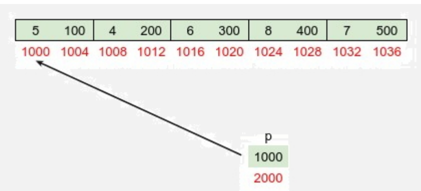
And one precedence rule we need to use:



Initial situation :

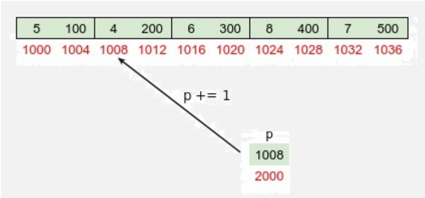


`struct test *p = st;`

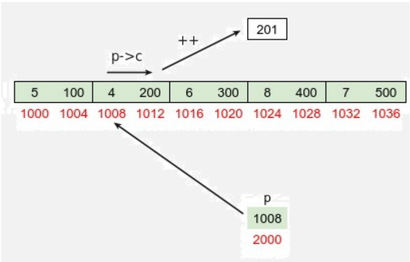


`p += 1;`

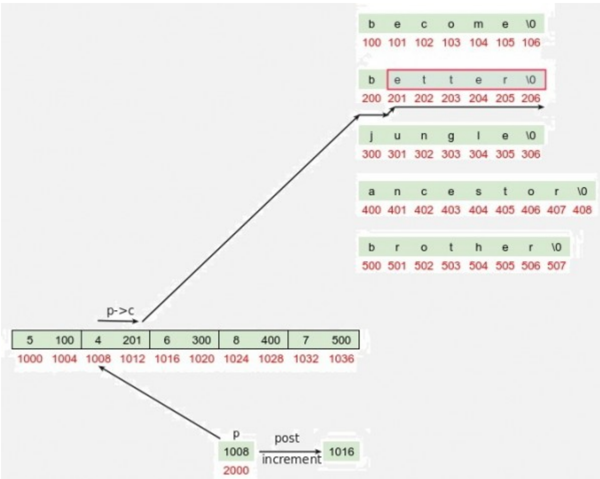
We know that if `ptr` is a pointer then, `ptr + x = ptr + x*sizeof(*ptr);`



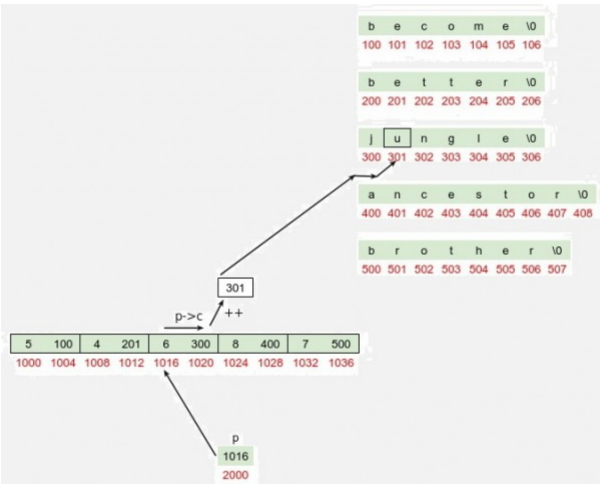

```
++p->c;
```



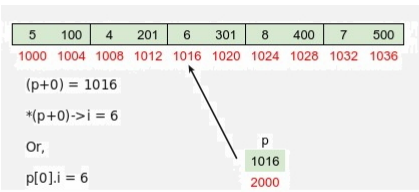

```
printf("%s,", p++->c);          // (p++)->c
```



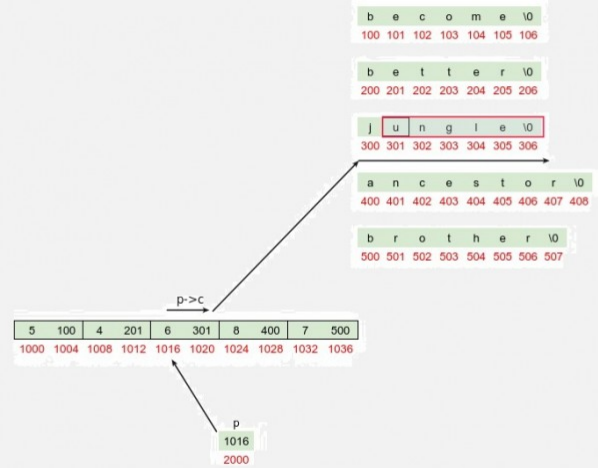

```
printf("%c,", *++p->c);      // *(++(p->c))
```




```
printf("%d,", p[0].i);
```



```
printf("%s \n", p->c);
```



Correct Answer: B

References

316 votes -- Debashish Deka (40.7k points)

4.13 Type Checking (1) top

4.13.1 Type Checking: GATE CSE 2003 | Question: 24 top https://gateoverflow.in/914

Which of the following statements is FALSE?

A. In statically typed languages, each variable in a program has a fixed type
B. In un-typed languages, values do not have any types
C. In dynamically typed languages, variables have no types
D. In all statically typed languages, each variable in a program is associated with values of only a single type during the execution of the program

gate2003-cse programming normal type-checking

Answer

Answers: Type Checking

4.13.1 Type Checking: GATE CSE 2003 | Question: 24 top https://gateoverflow.in/914

✔ Answer is (**C**). In dynamically typed languages variables do have type- just that it is inferred during runtime.

23 votes -- Arjun Suresh (330k points)

4.14 Union (1) top

4.14.1 Union: GATE CSE 2000 | Question: 1.17, ISRO2015-79 top https://gateoverflow.in/640

Consider the following C declaration:

```
struct (
    short x[5];
    union {
        float y;
        long z;
    } u;
)t;
```

Assume that the objects of the type short, float and long occupy 2 bytes, 4 bytes and 8 bytes, respectively. The memory requirement for variable t, ignoring alignment consideration, is:

A. 22 bytes
B. 14 bytes
C. 18 bytes
D. 10 bytes

gate2000-cse programming programming-in-c easy isro2015 union

Answer

Answers: Union

4.14.1 Union: GATE CSE 2000 | Question: 1.17, ISRO2015-79 top https://gateoverflow.in/640

✔ Correct Option: **C**

Here, structure creates the memory for 'array and union', but union only creates the memory for only 'long z' which is the largest size data type inside it.

Hence,

$short \times [5] = 5 * 2 = 10$ bytes [shorts take 2 bytes]

long z $= 8$ bytes

So, $(10 + 8) = 18$ bytes

53 votes -- Kalpna Bhargav (2.5k points)

4.15 Variable Binding (1) top

4.15.1 Variable Binding: GATE IT 2007 | Question: 34, UGCNET-Dec2012-III: 52 top https://gateoverflow.in/3467

Consider the program below in a hypothetical programming language which allows global variables and a choice of static or dynamic scoping.

```
int i;
program main()
{
    i = 10;
    call f();
}

procedure f()
{
    int i = 20;
    call g ();
}
procedure g()
{
    print i;
}
```

Let x be the value printed under static scoping and y be the value printed under dynamic scoping. Then, x and y are:

A. $x = 10, y = 20$
B. $x = 20, y = 10$
C. $x = 10, y = 10$
D. $x = 20, y = 20$

gate2007-it programming variable-binding normal ugcnetdec2012iii

Answer

Answers: Variable Binding

4.15.1 Variable Binding: GATE IT 2007 | Question: 34, UGCNET-Dec2012-III: 52 top https://gateoverflow.in/3467

✔ In static scoping, the scope of an identifier is determined by its location in the code, and since that doesn't change, the scope doesn't either. In dynamic scoping, the scope is determined by the sequence of calls that has led to the use of an identifier, and since that can be different each time that use is reached, is dynamic.

So, here:

Option A must be the answer.

As, under static scoping: $x = 10(\text{global } i)$

under dynamic scoping: $y = 20(\text{according to the sequence of calls,i.e } 20)$

20 votes -- sumit kumar singh dixit (1.6k points)

Answer Keys

4.1.1	A	4.2.1	C	4.2.2	C	4.2.3	C	4.2.4	2
4.2.5	6	4.2.6	19	4.2.7	C	4.2.8	C	4.2.9	B
4.2.10	C	4.2.11	B	4.3.1	A	4.3.2	A	4.4.1	N/A
4.4.2	C	4.4.3	3	4.4.4	B	4.5.1	N/A	4.5.2	N/A
4.5.3	N/A	4.5.4	N/A	4.5.5	A	4.5.6	B	4.5.7	C
4.5.8	C	4.6.1	N/A	4.6.2	C	4.6.3	B	4.6.4	D
4.6.5	B	4.6.6	D	4.6.7	6561	4.6.8	2016	4.6.9	30
4.6.10	A	4.6.11	D	4.6.12	C	4.7.1	D	4.7.2	D
4.7.3	A	4.7.4	A	4.7.5	C	4.7.6	D	4.7.7	140
4.7.8	D	4.7.9	D	4.8.1	B	4.9.1	A	4.9.2	B
4.9.3	B	4.9.4	B	4.9.5	D	4.9.6	C	4.9.7	D
4.9.8	A	4.9.9	D	4.9.10	C	4.9.11	C	4.9.12	D
4.9.13	B	4.9.14	D	4.9.15	-5	4.9.16	A	4.9.17	D
4.9.18	10	4.9.19	230	4.9.20	D	4.9.21	D	4.9.22	3
4.9.23	23	4.9.24	A	4.9.25	0	4.9.26	B	4.9.27	B
4.9.28	5	4.9.29	10	4.9.30	55	4.9.31	B	4.9.32	A
4.9.33	D	4.9.34	B	4.9.35	A	4.9.36	C	4.10.1	C
4.10.2	D	4.11.1	41	4.11.2	16	4.11.3	N/A	4.11.4	N/A
4.11.5	N/A	4.11.6	C	4.11.7	B	4.11.8	D	4.11.9	1.72 : 1.74
4.11.10	A	4.11.11	3	4.11.12	A	4.11.13	C	4.11.14	4
4.11.15	D	4.12.1	A	4.12.2	A	4.12.3	B	4.12.4	B
4.13.1	C	4.14.1	C	4.15.1	A				

5 Theory of Computation (270)

Regular expressions and finite automata, Context-free grammars and push-down automata, Regular and context-free languages, Pumping lemma, Turing machines and undecidability.

Mark Distribution in Previous GATE

Year	2021-1	2021-2	2020	2019	2018	2017-1	2017-2	2016-1	2016-2	Minimum	Average	Maximum
1 Mark Count	2	3	3	2	2	2	3	3	3	2	2.5	3
2 Marks Count	3	4	3	3	3	5	3	3	3	3	3.3	5
Total Marks	8	11	9	8	8	12	9	9	9	**8**	**9.2**	**12**

5.1 Closure Property (8) top

5.1.1 Closure Property: GATE CSE 1989 | Question: 3-ii top

https://gateoverflow.in/87117

Context-free languages and regular languages are both closed under the operation (s) of :

A. Union
B. Intersection
C. Concatenation
D. Complementation

gate1989 easy theory-of-computation closure-property multiple-selects

Answer

5.1.2 Closure Property: GATE CSE 1992 | Question: 16 top

https://gateoverflow.in/595

Which of the following three statements are true? Prove your answer.

i. The union of two recursive languages is recursive.
ii. The language $\{O^n \mid n \text{ is a prime}\}$ is not regular.
iii. Regular languages are closed under infinite union.

gate1992 theory-of-computation normal closure-property proof descriptive

Answer

5.1.3 Closure Property: GATE CSE 2002 | Question: 2.14 top

https://gateoverflow.in/844

Which of the following is true?

A. The complement of a recursive language is recursive
B. The complement of a recursively enumerable language is recursively enumerable
C. The complement of a recursive language is either recursive or recursively enumerable
D. The complement of a context-free language is context-free

gate2002-cse theory-of-computation easy closure-property

Answer

5.1.4 Closure Property: GATE CSE 2013 | Question: 17 top

https://gateoverflow.in/1439

Which of the following statements is/are **FALSE**?

1. For every non-deterministic Turing machine, there exists an equivalent deterministic Turing machine.
2. Turing recognizable languages are closed under union and complementation.
3. Turing decidable languages are closed under intersection and complementation.
4. Turing recognizable languages are closed under union and intersection.

A. 1 and 4 only
B. 1 and 3 only
C. 2 only
D. 3 only

gate2013-cse theory-of-computation normal closure-property

Answer

5.1.5 Closure Property: GATE CSE 2016 Set 2 | Question: 18 top

☛ https://gateoverflow.in/39574

Consider the following types of languages: L_1: Regular, L_2: Context-free, L_3: Recursive, L_4: Recursively enumerable. Which of the following is/are **TRUE ?**

I. $\overline{L_3} \cup L_4$ is recursively enumerable.
II. $\overline{L_2} \cup L_3$ is recursive.
III. $L_1^* \cap L_2$ is context-free.
IV. $L_1 \cup \overline{L_2}$ is context-free.

A. I only.
B. I and III only.
C. I and IV only.
D. I, II and III only.

gate2016-cse-set2 theory-of-computation regular-languages context-free-languages closure-property normal

Answer

5.1.6 Closure Property: GATE CSE 2017 Set 2 | Question: 04 top

☛ https://gateoverflow.in/118143

Let L_1, L_2 be any two context-free languages and R be any regular language. Then which of the following is/are CORRECT?

I. $L_1 \cup L_2$ is context-free
II. $\overline{L_1}$ is context-free
III. $L_1 - R$ is context-free
IV. $L_1 \cap L_2$ is context-free

A. I, II and IV only
B. I and III only
C. II and IV only
D. I only

gate2017-cse-set2 theory-of-computation closure-property

Answer

5.1.7 Closure Property: GATE CSE 2018 | Question: 7 top

☛ https://gateoverflow.in/204081

The set of all recursively enumerable languages is:

A. closed under complementation
B. closed under intersection
C. a subset of the set of all recursive languages
D. an uncountable set

gate2018-cse theory-of-computation closure-property easy

Answer

5.1.8 Closure Property: GATE IT 2006 | Question: 32 top

☛ https://gateoverflow.in/3571

Let L be a context-free language and M a regular language. Then the language $L \cap M$ is

A. always regular
B. never regular
C. always a deterministic context-free language
D. always a context-free language

gate2006-it theory-of-computation closure-property easy

Answer

Answers: Closure Property

5.1.1 Closure Property: GATE CSE 1989 | Question: 3-ii top

☛ https://gateoverflow.in/87117

✔ **Answer: (A) and (C)**

Regular language is closed under: **Union, Intersection, Concatenation, Complementation.**

CFL is closed under **Union Concatenation but not under Intersection or Complementation.**

So, both closed under Union Concatenation.

33 votes

-- Prashant Singh (47.1k points)

5.1.2 Closure Property: GATE CSE 1992 | Question: 16 top

☛ https://gateoverflow.in/595

✔

i. True. Recursive languages are closed under union.
ii. True. The language is context sensitive (we can write a C code right?) but not context-free (can be proved using pumping lemma for context-free languages).
iii. False. Regular languages are closed under finite union but not under infinite union.

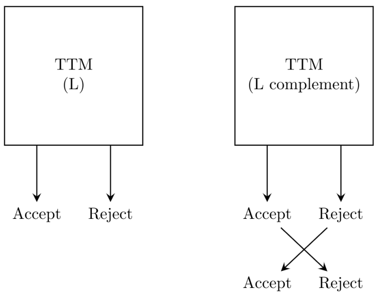
27 votes

-- Rajarshi Sarkar (27.8k points)

5.1.3 Closure Property: GATE CSE 2002 | Question: 2.14 top

☛ https://gateoverflow.in/844

✔



When a language is Recursive then there is a Total Turing machine means a Turing machine which have only two options either accept or rejects so if we complement a recursive language it works according to second figures hence it is recursive too.

For a recursively enumerable language there is Turing machine and so it has three options either accept, reject or loop. Hence its complement can be either recursively enumerable or not even recursively enumerable.

Among the options A is the best one though C is also not wrong as it includes the statement of A in an either clause.

Best Option: A

28 votes

-- Rishi yadav (9k points)

5.1.4 Closure Property: GATE CSE 2013 | Question: 17 top

☛ https://gateoverflow.in/1439

✔ Recursive enumerable languages are not closed under complement . while recursive languages are.

Both Recursive and Recursive enumerable languages are closed under intersection, union, and kleene star.

http://gatecse.in/wiki/Closure_Property_of_Language_Families

Non-Deterministic TM is equivalent to DTM

Only 2 is false. Option **C** is correct.

Note: Turing decidable language mean Recursive language and Turing recognizable language mean recursive enumerable language.

References

47 votes -- Praveen Saini (41.9k points)

5.1.5 Closure Property: GATE CSE 2016 Set 2 | Question: 18 top https://gateoverflow.in/39574

I. I. $\overline{L_3} \cup L_4$
L_3 is recursive, so $\overline{L_3}$ is also recursive (closed under complement),
So, $\overline{L_3}$ is recursive enumerable.
L_4 is recursive enumerable,
so, $\overline{L_3} \cup L_4$ is also recursive enumerable (closed under union).

II. $\overline{L_2} \cup L_3$
L_2 is Context-free, so $\overline{L_2}$, may or may not be Context-free (not closed under complement), but definitely $\overline{L_2}$ is Recursive.
L_3 is recursive.
so $\overline{L_2} \cup L_3$ is also recursive (closed under union).

III. $L_1^* \cap L_2$
L_1 is Regular, so L_1^* is also regular (closed under kleene star)
L_2 is Context-free
so, $L_1^* \cap L_2$ is also context-free (closed under intersection with regular).

IV. $L_1 \cup \overline{L_2}$
L_1 is regular.
L_2 is context-free, so $\overline{L_2}$ may or may not be Context-free (not closed under complement).
so, $L_1 \cup \overline{L_2}$ may or may not be Context-free.

Here, answer is **D**.

104 votes -- Praveen Saini (41.9k points)

5.1.6 Closure Property: GATE CSE 2017 Set 2 | Question: 04 top https://gateoverflow.in/118143

Statement I is **TRUE** as CFGs are closed under union.

Statement II is **FALSE** as CFGs are not enclosed under complementation.

Statement III is **TRUE** as L1−R can be written as $L1 \cap \bar{R}$. Regular language are closed under complementation and intersection of CFG and Regular is CFG.

Statement IV is **FALSE** as CFGs are not enclosed under intersection.

So, I and III are correct. Option B

29 votes -- Prateek Kumar (1.1k points)

5.1.7 Closure Property: GATE CSE 2018 | Question: 7 top https://gateoverflow.in/204081

Answer is **B.**

Reference: https://gatecse.in/closure-property-of-language-families/

C is false as the set of all recursively enumerable languages (semi-decidable) is a STRICT super set of the set of all recursive languages (decidable).

D is false as the set of all recursively enumerable languages (set of all Turing machines) is an infinite but countable set.

References

38 votes -- Arjun Suresh (330k points)

5.1.8 Closure Property: GATE IT 2006 | Question: 32 top https://gateoverflow.in/3571

✔ Let $\Sigma = \{a, b\}$.
$L_1 = \Sigma^*$ is a regular language
$L_2 = \{ww^R \mid w \in (a+b)^*\}$ is a context free language.
$L_1 \cap L_2 = \{ww^R \mid w \in (a+b)^*\}$ which is clearly context free language and not DCFL or Regular
Hence, the answer is option **D.**

45 votes -- Mari Ganesh Kumar (1.5k points)

5.2 Context Free Languages (33) top

5.2.1 Context Free Languages: GATE CSE 1987 | Question: 1-xii top https://gateoverflow.in/80291

A context-free grammar is ambiguous if:

A. The grammar contains useless non-terminals.
B. It produces more than one parse tree for some sentence.
C. Some production has two non terminals side by side on the right-hand side.
D. None of the above.

gate1987 theory-of-computation context-free-languages ambiguous

Answer

5.2.2 Context Free Languages: GATE CSE 1987 | Question: 2k top https://gateoverflow.in/80599

State whether the following statements are TRUE or FALSE:

The intersection of two CFL's is also a CFL.

gate1987 theory-of-computation context-free-languages true-false

Answer

5.2.3 Context Free Languages: GATE CSE 1992 | Question: 02,xix top https://gateoverflow.in/572

Context-free languages are:

A. closed under union
B. closed under complementation
C. closed under intersection
D. closed under Kleene closure

gate1992 context-free-languages theory-of-computation normal multiple-selects

Answer

5.2.4 Context Free Languages: GATE CSE 1992 | Question: 02,xviii top https://gateoverflow.in/576

If G is a context free grammar and w is a string of length l in $L(G)$, how long is a derivation of w in G, if G is in Chomsky normal form?

A. $2l$
B. $2l+1$
C. $2l-1$
D. l

gate1992 theory-of-computation context-free-languages easy multiple-selects

Answer

5.2.5 Context Free Languages: GATE CSE 1995 | Question: 2.20 top https://gateoverflow.in/2632

Which of the following definitions below generate the same language as L, where $L = \{x^n y^n \text{ such that } n \geq 1\}$?

I. $E \rightarrow xEy \mid xy$

II. $xy \mid (x^+xyy^+)$

III. x^+y^+

A. I only
B. I and II
C. II and III
D. II only

gate1995 theory-of-computation easy context-free-languages

Answer

5.2.6 Context Free Languages: GATE CSE 1996 | Question: 2.8 top https://gateoverflow.in/2737

If L_1 and L_2 are context free languages and R a regular set, one of the languages below is not necessarily a context free language. Which one?

A. $L_1 . L_2$
B. $L_1 \cap L_2$
C. $L_1 \cap R$
D. $L_1 \cup L_2$

gate1996 theory-of-computation context-free-languages easy

Answer

5.2.7 Context Free Languages: GATE CSE 1996 | Question: 2.9 top https://gateoverflow.in/2738

Define a context free languages $L \in \{0,1\}^*$, $\text{init}(L) = \{u \mid uv \in L$ for some v in $\{0,1\}^*\}$ (in other words, $\text{init}(L)$ is the set of prefixes of L)

Let $L = \{w \mid w$ is nonempty and has an equal number of 0's and 1's$\}$

Then $\text{init}(L)$ is:

A. the set of all binary strings with unequal number of 0's and 1's
B. the set of all binary strings including null string
C. the set of all binary strings with exactly one more 0 than the number of 1's or one more 1 than the number of 0's
D. None of the above

gate1996 theory-of-computation context-free-languages normal

Answer

5.2.8 Context Free Languages: GATE CSE 1999 | Question: 1.5 top https://gateoverflow.in/1459

Context-free languages are closed under:

A. Union, intersection
B. Union, Kleene closure
C. Intersection, complement
D. Complement, Kleene closure

gate1999 theory-of-computation context-free-languages easy

Answer

5.2.9 Context Free Languages: GATE CSE 1999 | Question: 7 top https://gateoverflow.in/1506

Show that the language

$$L = \{xcx \mid x \in \{0,1\}^* \text{ and } c \text{ is a terminal symbol}\}$$

is not context free. c is not 0 or 1.

gate1999 theory-of-computation context-free-languages normal proof

Answer

5.2.10 Context Free Languages: GATE CSE 2000 | Question: 7 top

https://gateoverflow.in/678

A. Construct as minimal finite state machine that accepts the language, over $\{0,1\}$, of all strings that contain neither the substring 00 nor the substring 11.
B. Consider the grammar
   - $S \to aSAb$
   - $S \to \epsilon$
   - $A \to bA$
   - $A \to \epsilon$

   where S, A are non-terminal symbols with S being the start symbol; a, b are terminal symbols and ϵ is the empty string. This grammar generates strings of the form $a^i b^j$ for some $i, j \geq 0$, where i and j satisfy some condition. What is the condition on the values of i and j?

gate2000-cse theory-of-computation descriptive regular-languages context-free-languages

Answer

5.2.11 Context Free Languages: GATE CSE 2001 | Question: 1.5 top

https://gateoverflow.in/698

Which of the following statements is true?

A. If a language is context free it can always be accepted by a deterministic push-down automaton
B. The union of two context free languages is context free
C. The intersection of two context free languages is a context free
D. The complement of a context free language is a context free

gate2001-cse theory-of-computation context-free-languages easy

Answer

5.2.12 Context Free Languages: GATE CSE 2003 | Question: 51 top

https://gateoverflow.in/940

Let $G = (\{S\}, \{a, b\}, R, S)$ be a context free grammar where the rule set R is $S \to aSb \mid SS \mid \epsilon$

Which of the following statements is true?

A. G is not ambiguous
B. There exist $x, y \in L(G)$ such that $xy \notin L(G)$
C. There is a deterministic pushdown automaton that accepts $L(G)$
D. We can find a deterministic finite state automaton that accepts $L(G)$

gate2003-cse theory-of-computation context-free-languages normal

Answer

5.2.13 Context Free Languages: GATE CSE 2005 | Question: 57 top

https://gateoverflow.in/1380

Consider the languages:

- $L_1 = \left\{ww^R \mid w \in \{0,1\}^*\right\}$
- $L_2 = \left\{w\#w^R \mid w \in \{0,1\}^*\right\}$, where $\#$ is a special symbol
- $L_3 = \{ww \mid w \in \{0,1\}^*\}$

Which one of the following is TRUE?

A. L_1 is a deterministic CFL
B. L_2 is a deterministic CFL
C. L_3 is a CFL, but not a deterministic CFL
D. L_3 is a deterministic CFL

gate2005-cse theory-of-computation context-free-languages easy

Answer

5.2.14 Context Free Languages: GATE CSE 2006 | Question: 19 top

https://gateoverflow.in/980

Let

$L_1 = \{0^{n+m}1^n0^m \mid n, m \geq 0\}$,

$L_2 = \{0^{n+m}1^{n+m}0^m \mid n, m \geq 0\}$ and

$L_3 = \{0^{n+m}1^{n+m}0^{n+m} \mid n, m \geq 0\}$.

Which of these languages are NOT context free?

A. L_1 only
B. L_3 only
C. L_1 and L_2
D. L_2 and L_3

gate2006-cse theory-of-computation context-free-languages normal

Answer

5.2.15 Context Free Languages: GATE CSE 2009 | Question: 12, ISRO2016-37 top

https://gateoverflow.in/1304

$$S \rightarrow aSa \mid bSb \mid a \mid b$$

The language generated by the above grammar over the alphabet $\{a, b\}$ is the set of:

A. all palindromes
B. all odd length palindromes
C. strings that begin and end with the same symbol
D. all even length palindromes

gate2009-cse theory-of-computation context-free-languages easy isro2016

Answer

5.2.16 Context Free Languages: GATE CSE 2015 Set 3 | Question: 32 top

https://gateoverflow.in/8489

Which of the following languages are context-free?

$L_1 : \{a^m b^n a^n b^m \mid m, n \geq 1\}$

$L_2 : \{a^m b^n a^m b^n \mid m, n \geq 1\}$

$L_3 : \{a^m b^n \mid m = 2n + 1\}$

A. L_1 and L_2 only
B. L_1 and L_3 only
C. L_2 and L_3 only
D. L_3 only

gate2015-cse-set3 theory-of-computation context-free-languages normal

Answer

5.2.17 Context Free Languages: GATE CSE 2016 Set 1 | Question: 16 top

https://gateoverflow.in/39640

Which of the following languages is generated by the given grammar?

$$S \rightarrow aS \mid bS \mid \varepsilon$$

A. $\{a^n b^m \mid n, m \geq 0\}$
B. $\{w \in \{a, b\}^* \mid w \text{ has equal number of a's and b's}\}$
C. $\{a^n \mid n \geq 0\} \cup \{b^n \mid n \geq 0\} \cup \{a^n b^n \mid n \geq 0\}$
D. $\{a, b\}^*$

gate2016-cse-set1 theory-of-computation context-free-languages normal

Answer

5.2.18 Context Free Languages: GATE CSE 2016 Set 1 | Question: 42 top

https://gateoverflow.in/39705

Consider the following context-free grammars;

$G_1 : S \to aS \mid B, B \to b \mid bB$

$G_2 : S \to aA \mid bB, A \to aA \mid B \mid \varepsilon, B \to bB \mid \varepsilon$

Which one of the following pairs of languages is generated by G_1 and G_2 ,respectively?

A. $\{a^m b^n \mid m > 0 \text{ or } n > 0\}$ and $\{a^m b^n \mid m > 0 \text{ and } n > 0\}$
B. $\{a^m b^n \mid m > 0 \text{ and } n > 0\}$ and $\{a^m b^n \mid m > 0 \text{ or } n \geq 0\}$
C. $\{a^m b^n \mid m \geq 0 \text{ or } n > 0\}$ and $\{a^m b^n \mid m > 0 \text{ and } n > 0\}$
D. $\{a^m b^n \mid m \geq 0 \text{ and } n > 0\}$ and $\{a^m b^n \mid m > 0 \text{ or } n > 0\}$

gate2016-cse-set1 theory-of-computation context-free-languages normal

Answer

5.2.19 Context Free Languages: GATE CSE 2016 Set 2 | Question: 43 top

https://gateoverflow.in/39605

Consider the following languages:

$L_1 = \{a^n b^m c^{n+m} : m, n \geq 1\}$

$L_2 = \{a^n b^n c^{2n} : n \geq 1\}$

Which one of the following is TRUE?

A. Both L_1 and L_2 are context-free.
B. L_1 is context-free while L_2 is not context-free.
C. L_2 is context-free while L_1 is not context-free.
D. Neither L_1 nor L_2 is context-free.

gate2016-cse-set2 theory-of-computation context-free-languages normal

Answer

5.2.20 Context Free Languages: GATE CSE 2017 Set 1 | Question: 10 top

https://gateoverflow.in/118290

Consider the following context-free grammar over the alphabet $\Sigma = \{a, b, c\}$ with S as the start symbol:

$$S \to abScT \mid abcT$$

$$T \to bT \mid b$$

Which one of the following represents the language generated by the above grammar?

A. $\{(ab)^n (cb)^n \mid n \geq 1\}$
B. $\{(ab)^n cb^{m_1} cb^{m_2} \dots cb^{m_n} \mid n, m_1, m_2, \dots, m_n \geq 1\}$
C. $\{(ab)^n (cb^m)^n \mid m, n \geq 1\}$
D. $\{(ab)^n (cb^n)^m \mid m, n \geq 1\}$

gate2017-cse-set1 theory-of-computation context-free-languages normal

Answer

5.2.21 Context Free Languages: GATE CSE 2017 Set 1 | Question: 34 top

https://gateoverflow.in/118316

If G is a grammar with productions

$S \to SaS \mid aSb \mid bSa \mid SS \mid \epsilon$

where S is the start variable, then which one of the following strings is not generated by G?

A. $abab$
B. $aaab$

C. $abbaa$
D. $babba$

gate2017-cse-set1 theory-of-computation context-free-languages normal

Answer

5.2.22 Context Free Languages: GATE CSE 2017 Set 1 | Question: 38 top https://gateoverflow.in/118321

Consider the following languages over the alphabet $\sum = \{a, b, c\}$. Let $L_1 = \{a^n b^n c^m | m, n \geq 0\}$ and $L_2 = \{a^m b^n c^n | m, n \geq 0\}$.

Which of the following are context-free languages?

I. $L_1 \cup L_2$
II. $L_1 \cap L_2$

A. I only
B. II only
C. I and II
D. Neither I nor II

gate2017-cse-set1 theory-of-computation context-free-languages normal

Answer

5.2.23 Context Free Languages: GATE CSE 2017 Set 2 | Question: 16 top https://gateoverflow.in/118243

Identify the language generated by the following grammar, where S is the start variable.

- $S \rightarrow XY$
- $X \rightarrow aX \mid a$
- $Y \rightarrow aYb \mid \epsilon$

A. $\{a^m b^n \mid m \geq n, n > 0\}$
B. $\{a^m b^n \mid m \geq n, n \geq 0\}$
C. $\{a^m b^n \mid m > n, n \geq 0\}$
D. $\{a^m b^n \mid m > n, n > 0\}$

gate2017-cse-set2 theory-of-computation context-free-languages

Answer

5.2.24 Context Free Languages: GATE CSE 2019 | Question: 31 top https://gateoverflow.in/302817

Which one of the following languages over $\Sigma = \{a, b\}$ is NOT context-free?

A. $\{ww^R \mid w \in \{a, b\}^*\}$
B. $\{wa^n b^n w^R \mid w \in \{a, b\}^*, n \geq 0\}$
C. $\{wa^n w^R b^n \mid w \in \{a, b\}^*, n \geq 0\}$
D. $\{a^n b^i \mid i \in \{n, 3n, 5n\}, n \geq 0\}$

gate2019-cse theory-of-computation context-free-languages

Answer

5.2.25 Context Free Languages: GATE CSE 2021 Set 1 | Question: 1 top https://gateoverflow.in/357451

Suppose that L_1 is a regular language and L_2 is a context-free language. Which one of the following languages is NOT necessarily context-free?

A. $L_1 \cap L_2$
B. $L_1 \cdot L_2$
C. $L_1 - L_2$
D. $L_1 \cup L_2$

gate2021-cse-set1 context-free-languages theory-of-computation

Answer

5.2.26 Context Free Languages: GATE CSE 2021 Set 2 | Question: 41 top https://gateoverflow.in/357499

For a string w, we define w^R to be the reverse of w. For example, if $w = 01101$ then $w^R = 10110$.

Which of the following languages is/are context-free?

A. $\{wxw^Rx^R \mid w, x \in \{0,1\}^*\}$
B. $\{ww^Rxx^R \mid w, x \in \{0,1\}^*\}$
C. $\{wxw^R \mid w, x \in \{0,1\}^*\}$
D. $\{wxx^Rw^R \mid w, x \in \{0,1\}^*\}$

gate2021-cse-set2 multiple-selects theory-of-computation context-free-languages

Answer

5.2.27 Context Free Languages: GATE IT 2006 | Question: 34 top https://gateoverflow.in/3573

In the context-free grammar below, S is the start symbol, a and b are terminals, and ϵ denotes the empty string.

- $S \to aSAb \mid \epsilon$
- $A \to bA \mid \epsilon$

The grammar generates the language

A. $((a+b)^*b)$
B. $\{a^mb^n \mid m \leq n\}$
C. $\{a^mb^n \mid m = n)$
D. a^*b^*

gate2006-it theory-of-computation context-free-languages normal

Answer

5.2.28 Context Free Languages: GATE IT 2006 | Question: 4 top https://gateoverflow.in/3543

In the context-free grammar below, S is the start symbol, a and b are terminals, and ϵ denotes the empty string

$S \to aSa \mid bSb \mid a \mid b \mid \epsilon$

Which of the following strings is NOT generated by the grammar?

A. $aaaa$
B. $baba$
C. $abba$
D. $babaaabab$

gate2006-it theory-of-computation context-free-languages easy

Answer

5.2.29 Context Free Languages: GATE IT 2007 | Question: 46 top https://gateoverflow.in/3481

The two grammars given below generate a language over the alphabet $\{x, y, z\}$

$G1 : S \to x \mid z \mid x\ S \mid z\ S \mid y\ B$
$\quad\ B \to y \mid z \mid y\ B \mid z\ B$

$G2 : S \to y \mid z \mid y\ S \mid z\ S \mid x\ B$
$\quad\ B \to y \mid y\ S$

Which one of the following choices describes the properties satisfied by the strings in these languages?

A. $G1$: No y appears before any x
$G2$: Every x is followed by at least one y
B. $G1$: No y appears before any x
$G2$: No x appears before any y

C. $G1$: No y appears after any x
$G2$: Every x is followed by at least one y
D. $G1$: No y appears after any x
$G2$: Every y is followed by at least one x

gate2007-it theory-of-computation normal context-free-languages

Answer

5.2.30 Context Free Languages: GATE IT 2007 | Question: 48 top

https://gateoverflow.in/3490

Consider the grammar given below:

$S \to x\ B \mid y\ A$
$A \to x \mid x\ S \mid y\ A\ A$
$B \to y \mid y\ S \mid x\ B\ B$

Consider the following strings.

i. $xxyyx$
ii. $xxyyxy$
iii. $xyxy$
iv. $yxxy$
v. yxx
vi. xyx

Which of the above strings are generated by the grammar ?

A. i, ii and iii
B. ii, v and vi
C. ii, iii and iv
D. i, iii and iv

gate2007-it theory-of-computation context-free-languages normal

Answer

5.2.31 Context Free Languages: GATE IT 2007 | Question: 49 top

https://gateoverflow.in/3491

Consider the following grammars. Names representing terminals have been specified in capital letters.

$G1$:	stmnt	$\to$	WHILE (expr) stmnt			
	stmnt	$\to$	OTHER			
	expr	$\to$	ID			
$G2$:	stmnt	$\to$	WHILE (expr) stmnt			
	stmnt	$\to$	OTHER			
	expr	$\to$		expr	$+$	expr
	expr	$\to$		expr	$*$	expr
	expr	$\to$	ID			

Which one of the following statements is true?

A. G_1 is context-free but not regular and G_2 is regular
B. G_2 is context-free but not regular and G_1 is regular
C. Both G_1 and G_2 are regular
D. Both G_1 and G_2 are context-free but neither of them is regular

gate2007-it theory-of-computation context-free-languages normal

Answer

5.2.32 Context Free Languages: GATE IT 2008 | Question: 34 top

https://gateoverflow.in/3344

Consider a CFG with the following productions.

$S \rightarrow AA \mid B$
$A \rightarrow 0A \mid A0 \mid 1$
$B \rightarrow 0B00 \mid 1$

S is the start symbol, A and B are non-terminals and 0 and 1 are the terminals. The language generated by this grammar is:

A. $\{0^n 10^{2n} \mid n \geq 1\}$
B. $\{0^i 10^j 10^k \mid i, j, k \geq 0\} \cup \{0^n 10^{2n} \mid n \geq 0\}$
C. $\{0^i 10^j \mid i, j \geq 0\} \cup \{0^n 10^{2n} \mid n \geq 0\}$
D. The set of all strings over $\{0, 1\}$ containing at least two 0's

gate2008-it theory-of-computation context-free-languages normal

Answer

5.2.33 Context Free Languages: GATE IT 2008 | Question: 78 top

https://gateoverflow.in/3392

A CFG G is given with the following productions where S is the start symbol, A is a non-terminal and a and b are terminals.

- $S \rightarrow aS \mid A$
- $A \rightarrow aAb \mid bAa \mid \epsilon$

Which of the following strings is generated by the grammar above?

A. $aabbaba$
B. $aabaaba$
C. $abababb$
D. $aabbaab$

gate2008-it theory-of-computation normal context-free-languages

Answer

Answers: Context Free Languages

5.2.1 Context Free Languages: GATE CSE 1987 | Question: 1-xii top

https://gateoverflow.in/80291

✔ An ambiguous grammar produces more than one parse tree for any string.

Correct Answer : B.

24 votes -- kirti singh (2.6k points)

5.2.2 Context Free Languages: GATE CSE 1987 | Question: 2k top

https://gateoverflow.in/80599

✔ No intersection of two CFLs may or may not be a CFL i.e. CFL is not closed under intersection operation.

Example:

- $L_1 : \{a^n b^n c^m | n, m >= 1\} \cap L_2 : \{a^n b^m c^m | n, m >= 1\}$
- $L_3 = \{a^m b^m c^m | m >= 1\}$, which is CSL.

25 votes -- Prashant Singh (47.1k points)

5.2.3 Context Free Languages: GATE CSE 1992 | Question: 02,xix top

https://gateoverflow.in/572

✔ Answer: **A;D.**

Context-Free languages are not closed under intersection and complementation.

23 votes -- Rajarshi Sarkar (27.8k points)

5.2.4 Context Free Languages: GATE CSE 1992 | Question: 02,xviii top https://gateoverflow.in/576

✔ Chomsky Normal Form (If all of its production rules are of the form):

- $A \to BC$ or
- $A \to a$ or
- $S \to \varepsilon$

where A, B and C are nonterminal symbols, a is a terminal symbol (a symbol that represents a constant value), S is the start symbol, and ε is the empty string. Also, neither B nor C may be the start symbol, and the third production rule can only appear if ε is in $L(G)$, namely, the language produced by the context-free grammar G.

Applying productions of the first form will increase the number of nonterminals from k to $k+1$, since you replace one nonterminal (-1) with two nonterminals $(+2)$ for a net gain of $+1$ nonterminal. Since you start with one nonterminal, this means you need to do $l-1$ productions of the first form. You then need l more of the second form to convert the nonterminals to terminals, giving a total of $l+(l-1)=2l-1$ productions.

Correct Answer: C

43 votes -- Rajarshi Sarkar (27.8k points)

5.2.5 Context Free Languages: GATE CSE 1995 | Question: 2.20 top https://gateoverflow.in/2632

✔ Correct Option: **A**

In the other two you can have any number of x and y. There is no such restriction over the number of both being equal.

31 votes -- Gate Keeda (15.9k points)

5.2.6 Context Free Languages: GATE CSE 1996 | Question: 2.8 top https://gateoverflow.in/2737

✔ Correct Option: **B**

CFL's are not closed under intersection.

23 votes -- Gate Keeda (15.9k points)

5.2.7 Context Free Languages: GATE CSE 1996 | Question: 2.9 top https://gateoverflow.in/2738

✔ Correct Option: **B**

Because for any binary string of 0's and 1's we can append another string to make it contain equal number of 0's and 1's. i.e., any string over $\{0,1\}$ is a prefix of a string in L.

Example:

01111 - is prefix of 01111000 which is in L.
1111- is prefix of 11110000 which is in L.
01010- is prefix of 010101 which is in L.

39 votes -- Arjun Suresh (330k points)

5.2.8 Context Free Languages: GATE CSE 1999 | Question: 1.5 top https://gateoverflow.in/1459

✔ **Context free languages** are not closed under intersection and complement. Correct option is **(B)** Union and Kleene closure.

18 votes -- Bhagirathi Nayak (11.7k points)

5.2.9 Context Free Languages: GATE CSE 1999 | Question: 7 top https://gateoverflow.in/1506

✔ We can prove that the given language L is Not CFL using the "Pumping lemma for CFLs". That is formal and correct way to prove this. Since this Question is subjective, and that year GATE was a subjective exam, so All other answers which give some vague informal idea (I have written that informal idea at the end of the answer) behind L being Non-CFL would have been awarded zero marks as that doesn't prove anything.

Pumping lemma for CFLs:

Let L be a CFL. Then there exists some integer constant $P \geq 1$ (Called Pumping length or pumping-lemma constant) such that if $w \in L$ with $|w| \geq P$, then we can write $w = uvxyz$, subject to the following conditions:

1. $|vxy| \leq P$.

2. $vy \neq \in$.

3. For all $i \geq 0$, we have $uv^ixy^iz \in L$.

i.e. Informally, For every sufficiently large string w in L, We must be able to split it such that it is possible to find at most two short, nearby substrings that we can “pump” i times in tandem, for any non-negative integer i, and the resulting string will still be in that language.

- So, In the Whole String w with $|w| \geq P$, "Anywhere in this string", "Within At Most P symbols", We must find Two sub-strings(Possibly empty, Not Both Though) such that we can "Pump" both of these sub-strings $i \geq 0$ times in tandem i.e. Both repeated $i \geq 0$ times.

Now, Assume that Given L is CFL, Hence, It will satisfy Pumping lemma for CFL.

So, There must be some positive integer constant (pumping length) ≥ 1 existing for this language. Let it be P.

So, Now for every string $w \in L$ whose length is greater than or equal to P, we must have some partition $uvxyz$ satisfying all the above conditions.

So, Let me take the string $1^P0^Pc1^P0^P \in L$, Now Try to split it into five parts $uvxyz$ such that All the Three conditions of pumping lemma must satisfy.

Basically in Pumping lemma for CFL, You want to find "at most P" consecutive symbols in the String(anywhere in the String) such that you can find two short sub-strings in that part and Pump those sub-strings in tandem.

But for the above String $1^P0^Pc1^P0^P$, we cannot find any such at most P consecutive symbols anywhere in the string which will satisfy the Pumping lemma conditions. (Hint : Check different possibilities i.e. Take Those at most P consecutive symbols completely in 1^P part or completely in 0^P part or some in 0^P part and some in 1^P part or some in 1^P part and some in 0^P part of the string .. etc.. covering all possible partitions..)

When you take vxy substring completely in 1^P part and however you choose v, y in this part, one thing is certain that when you repeat v, y Zero Times then at least one 1 will be removed from the string and hence the resulting string will not belong to the language because the substring to the left and right of c will not remain the same. Similar logic applies for when you take vxy substring completely in 0^P part.

When you take vxy substring between $1's$ and $0's$, however you choose v, y in this part, one thing is certain that when you repeat v, y Zero Times then at least one 1 and at least one 0 will be removed from the string and hence the resulting string will not belong to the language because the substring to the left and right of c will not remain the same.

So, Given language doesn't satisfy Pumping lemma and hence the contradiction. So, L is Not CFL.

For more examples to understand Pumping lemma, I have answered many Questions based on Pumping lemma and You can refer to them in my profile under "All Answers" tab.

The Informal idea behind L being Non-CFL is that in PDA, we have a Stack as an auxiliary memory. And Stack works in LIFO manner. Since we have the language as w_1cw_2 So, if you want to match the substring to the left of c i.e. w_1 with that of to the right of c i.e. w_2, You need to PUSH left substring w_1 into the stack and skip c and now You need to match the first symbol of w_2 with the first symbol of w_1 But the first symbol of w_1 is at the bottom of the Stack. So, that's the problem with Stack. Hence, we can informally say that Forward matching cannot be done by Stack or for that matter by PDA.

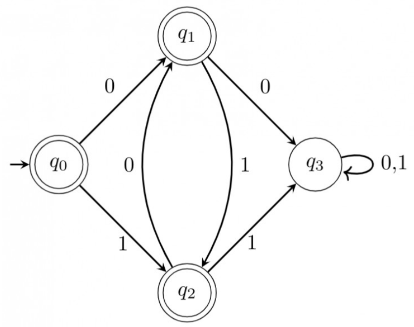
23 votes -- Deepak Poonia (23.3k points)

5.2.10 Context Free Languages: GATE CSE 2000 | Question: 7 top

https://gateoverflow.in/678

✔

a. Language $L = (0+1)^* - (0+1)^*(00+11)(0+1)^*$
   DFA contains 4 states of which 3 are final and 1 is dead state.



b. $i \leq j$
as $S \to aSAb$
There will be always one a in left and minimum one b in right and $A \to bA \mid \epsilon$ can generate any number of b's including null string. If A is ϵ then $i = j$ and if A is generating any b, then $j > i$ so condition is $i \leq j$.

35 votes -- Mithlesh Upadhyay (4.3k points)

5.2.11 Context Free Languages: GATE CSE 2001 | Question: 1.5 top https://gateoverflow.in/698

✔ Answer is (**B**).

(A) is wrong as a language can be context free even if it is being accepted by non-deterministic PDA for ex- $\{ww^r : w \in \Sigma^*$ and w^r is reverse of $w\}$

(C) and (D) not always true as Context free languages are not closed under Complement or Intersection.

24 votes -- Prateeksha Keshari (1.7k points)

5.2.12 Context Free Languages: GATE CSE 2003 | Question: 51 top https://gateoverflow.in/940

✔ It will be easy to analyze the problem if we replace terminal a and b by **(** and **)** respectively.

$S \to (S) \mid SS \mid \epsilon$

$L(G) =$ balanced parentheses [each left parenthesis has a matching right parenthesis and are well nested]

example $(), ()(), (()), (()()())$.

A. $S \Rightarrow (S) \Rightarrow ()$
$S \Rightarrow SS \Rightarrow S \Rightarrow (S) \Rightarrow ()$
$S \Rightarrow SS \Rightarrow S \Rightarrow (S) \Rightarrow ()$
String $()$ can be derived by above three way each having different derivation tree.
So G is Ambiguous
B. Concatenation of two balance parenthesis will be balanced also . eq. $x = (())$ $y = ()$ $xy = (())()$.
C. We can design Deterministic PDA for L. put left parenthesis (only) in stack and pop with right parenthesis.
D. We cannot design DFA for L because we need a stack to match left parenthesis with right parenthesis.

only option **C** is true.

77 votes -- Praveen Saini (41.9k points)

5.2.13 Context Free Languages: GATE CSE 2005 | Question: 57 top https://gateoverflow.in/1380

✔ Correct Option: **B**

Reference: http://gatecse.in/wiki/Identify_the_class_of_the_language

References

20 votes -- Gate Keeda (15.9k points)

5.2.14 Context Free Languages: GATE CSE 2006 | Question: 19 top

https://gateoverflow.in/980

✔ **Answer is (D)**

L_1 is context-free. We count the number of 0's and check if the remaining number of 1's followed by 0's count to the initial number of 0's.

L_2 is not context-free. Here the number of 0's and the following 1's must be same, which can be checked using a PDA. But after that we must also ensure that the following number of 0's must be less than the previous count of 0's and 1's (otherwise $n < 0$, which violates the condition for acceptance) and we cannot do these two checks using a single PDA.

L_3 is again not context-free as it is nothing but equal number of 0's followed by equal number of 1's followed by equal number of 0's.

51 votes

-- Arjun Suresh (330k points)

5.2.15 Context Free Languages: GATE CSE 2009 | Question: 12, ISRO2016-37 top

https://gateoverflow.in/1304

✔ Answer is **B**. String generated by this language is a,b,aba,bab,aabaa,.....

All this strings are odd length palindromes.

31 votes

-- neha pawar (3.3k points)

5.2.16 Context Free Languages: GATE CSE 2015 Set 3 | Question: 32 top

https://gateoverflow.in/8489

✔ first check for L_1. now look a^m & b^m and a^n & b^n must be comparable using one stack for CFL.
now take a stack push all a^m in to the stack then push all b^n in to stack now a^n is coming so pop b^n for each a^n by this b^n and a^n will b comparable. now we have left only a^m in stack and b^m is coming so pop a^m for each b^m by which we can compare a^m to b^m ..we conclude that we are comparing this L_1 using a single stack so this is CFG.

now for L_2.this can not be done in to a single stack because m and n are not comparable we can not find when to push or pop so this is CSL.

now for L_3.push all a's into stack and pop $2a$'s for every b and at last we left with a single a .
bcz here $aaaaabb$ is a valid string where $m = 2n + 1$ and $n = 2$. So realized using single stack hence L_3 is CFG.

so the option is **B**.. L_1 and L_3 are CFG

54 votes

-- Anoop Sonkar (4.1k points)

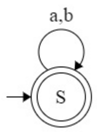
5.2.17 Context Free Languages: GATE CSE 2016 Set 1 | Question: 16 top

https://gateoverflow.in/39640

✔ Correct Option: **D**

$S \rightarrow aS \mid bS \mid \varepsilon$

is $(a+b)^*$



58 votes

-- Praveen Saini (41.9k points)

5.2.18 Context Free Languages: GATE CSE 2016 Set 1 | Question: 42 top

https://gateoverflow.in/39705

✔ Answer is (**D**).

G_1 results in strings $b, ab, bb, aab, abb, bbb,$ ... i.e $a^m b^n$, $m \geq 0$ **and** $n > 0$ (and because only a's are not possible but only b's are)

G_2 result in strings $a, b, aa, ab, bb, aaa, aab, abb, bbb$ ... i.e $a^m b^n$, $m > 0$ **or** $n > 0$ (or because only b's is possible $b, bb, bbb,$, only a's are possible)

72 votes

-- juxtapose (393 points)

5.2.19 Context Free Languages: GATE CSE 2016 Set 2 | Question: 43 top https://gateoverflow.in/39605

✔ $L_1 = \{a^n b^m c^{n+m} \ : \ m, n \geq 1\}$ is Context-free language

(push $a's$ into stack, then push $b's$ into stack , read $c's$ and pop $b's$, when no $b's$ left on stack, keep reading $c's$ and pop $a's$, when no c's left in input , and stack is empty, then accepted).

$L_2 = \{a^n b^n c^{2n} \ : \ n \geq 1\}$ is Context-sensitive language and not context-free. (cannot implemented by one stack)

So, answer is option **B.**

55 votes -- Praveen Saini (41.9k points)

5.2.20 Context Free Languages: GATE CSE 2017 Set 1 | Question: 10 top https://gateoverflow.in/118290

✔ **Answer should be B**

Consider the 1st production $S \to abScT$

This production generates equal number of (ab)'s and c's but after each c there is T which goes to $T \to bT$

So, with each c there can be one or more b's (one because of production $T \to b$ and more because of $T \to bT$)

and these b's are independent.

For example,

$ababcbbcbb$ is the part of the language

and $ababcbbbbbbcbb$ is also the part of the language so we can rule out options A and C as both say equal number of b's after each c.

In option D equal number of (ab)'s and c's is not satisfied. The only option that satisfies these 2 conditions is option B.

57 votes -- (points)

5.2.21 Context Free Languages: GATE CSE 2017 Set 1 | Question: 34 top https://gateoverflow.in/118316

✔ $S \to SaS \mid aSb \mid bSa \mid SS \mid \epsilon$

If we observe carefully the given grammar is generating all strings over $\Sigma = \{a, b\}$ having number of $b's$ not exceeding the number of $a's$. So, we can straight away give answer as option D. Strings in other options can be generated as shown below.

A. $abab$
$S \to a\boxed{S}b$
$\to ab\boxed{S}ab \to abab$

B. $aaab$
$S \to \boxed{S}aS$
$\to \boxed{S}aSaS$
$\to a\boxed{S}aS$
$\to aa\boxed{S}$
$\to aaa\boxed{S}b$
$\to aaab$

C. $abbaa$
$S \to \boxed{S}S$
$\to a\boxed{S}bS$
$\to ab\boxed{S}$
$\to abb\boxed{S}a$
$\to abb\boxed{S}aSa$
$\to abba\boxed{S}a$
$\to abbaa$

Hence strings in options A, B and C can be generated using the given grammar but not the one in option D. Answer is D.

34 votes -- Arnabi Bej (5.8k points)

5.2.22 Context Free Languages: GATE CSE 2017 Set 1 | Question: 38 top

https://gateoverflow.in/118321

✔ Correct Option: **A**

1. Here, let's say $L = L_1 \cup L_2$.
   Here as we see , L_1 and L_2 both are DCFLs. And hence CFLs.
   And from the Closure Properties of CFLs, we can conclude that CFLs are Closed under Union Operation.
   And hence, $L = L_1 \cup L_2$ is CFL.

2. Now, let's take $L = L_1 \cap L_2$,and that can be expressed as,
   $L = \{a^i b^j c^k \mid i = j \ AND \ j = k, i, j, k \geq 0\}$, in other words
   $L = \{a^n b^n c^n \mid n >= 0\}$
   And we know that it comes under CSL (As we will require Two Stacks at a Time). So L is CSL and also NOT a CFL.

43 votes -- Tilak D. Nanavati (2.9k points)

5.2.23 Context Free Languages: GATE CSE 2017 Set 2 | Question: 16 top

https://gateoverflow.in/118243

✔ $S \rightarrow XY$

$X \rightarrow aX \mid a$

$Y \rightarrow aYb \mid \epsilon$

X generates atleast one 'a'. While Y generates equal no of a's and b's(including epsilon).

$L = \{a, aa, aaa, aab, aaaa, aaab, aaaaa, aaabb, \ldots\}$

Hence, answer should be Option C.

21 votes -- tvkkk (1.1k points)

5.2.24 Context Free Languages: GATE CSE 2019 | Question: 31 top

https://gateoverflow.in/302817

✔ **Answer : C**

Input alphabet $\Sigma = \{a, b\}$

A.

$\{ww^R | w \in \{a, b\}^*\}$

This language is well-known CFL. The CFG generating this language is as following :

$G_1 : S \rightarrow aSa | bSb | \in$

B.

$\{wa^n b^n w^R | w \in \{a, b\}^*, n \geq 0\}$

This language is also CFL and can easily be generated by a slight modification in the above Grammar G_1. The CFG generating this language is as following :

$G_2 : S \rightarrow aSa | bSb | \in | A$

$A \rightarrow aAb | \in$

D.

$\{a^n b^i | i \in \{n, 3n, 5n\}, n \geq 0\}$

This language is also CFL. It can be seen as Union of Three CFLs $a^n b^n, a^n b^{3n}, a^n b^{5n}$. . The CFG generating this language is as following :

$G : S \rightarrow A | B | C$

$A \rightarrow aAb | \in$

$B \rightarrow aBbbb | \in$

$C \rightarrow aCbbbbb | \in$

So, By now, you could get the answer to be the Third Option i.e. C Or we can use Pumping lemma to show that It is Non-CFL.

C.

$L = \{wa^n w^R b^n | w \in \{a,b\}^*, n \geq 0\}$

L is a Non-CFL.

We can use "Pumping lemma for CFL" to prove that L is Not a CFL.

Pumping lemma for CFLs:

Let L be a CFL. Then there exists some integer constant $P \geq 1$ (Called Pumping length or pumping-lemma constant) such that if $w \in L$ with $|w| \geq P$, then we can write $w = uvxyz$, subject to the following conditions:

1. $|vxy| \leq P$.

2. $vy \neq \in$.

3. For all $i \geq 0$, we have $uv^i xy^i z \in L$.

i.e. Informally, For every sufficiently large string w in L, We must be able to split it such that it is possible to find at most two short, nearby substrings that we can "pump" i times in tandem, for any non-negative integer i, and the resulting string will still be in that language.

Now, Assume that Given L is CFL, Hence, It will satisfy Pumping lemma for CFL.

So, There must be some integer constant (pumping length) ≥ 1 exists for this language. Let it be P.

So, Now for every string $w \in L$ whose length is greater than or equal to P, we must have some partition $uvxyz$ satisfying all the above conditions.

So, Let me take the string $b^{2P}a^{2P}b^{2P}b^{2P}$, Now Try to split it into five parts $uvxyz$ such that All the Three conditions of pumping lemma must satisfy.

Basically in Pumping lemma for CFL, You want to find "at most P" consecutive symbols in the String(anywhere in the String) such that you can find two short sub-strings in that part and Pump those sub-strings in tandem.

But for the above String $b^{2P}a^{2P}b^{2P}b^{2P}$, we cannot find any such at most P consecutive symbols anywhere in the string which will satisfy the Pumping lemma conditions. (Hint : Take Those at most P consecutive symbols in the first part of the string i.e. b^{2P} or in the second part i.e. a^{2P} or in third part or in fourth part or between the parts etc.. covering all possible partitions..)

So, Given language doesn't satisfy Pumping lemma and hence, L is Not CFL.

The informal/intuitive idea for Non-CFLness of L (language in Option C) is that You need PDA to do the reverse matching i.e. $w \; matched \; with \; w^R$ But Because a^n is in between $w \; and \; w^R$ and b^n is after w^R, You can either match ww^R or you can match $a^n b^n$ but Not both simultaneously. This is Informal Idea and by practice It becomes easier to check for a language being CFL or Not using this informal method.

45 votes -- Deepak Poonia (23.3k points)

5.2.25 Context Free Languages: GATE CSE 2021 Set 1 | Question: 1 top https://gateoverflow.in/357451

✔ Correct Option: C

Given,

- $L_1 -$ Regular Language (RL)
- $L_2 -$ Context-Free Language (CFL)

(A) $L_1 \cap L_2 \rightarrow$ **CFL**

Because **intersection operation** with regular languages is **closed under CFLs**.

Hence, **True**.

(B) $L_1 . L_2 \rightarrow$ **CFL**

Every **regular language** is a **CFL** and **CFLs** are closed under **concatenation**.

Hence, **True**.

(C) $L_1 - L_2 \equiv L_1 \cap L_2^c$.

Suppose, let's consider $L_1 = \Sigma^*$ and L_2 as any **CFL** and we get $L_1 \cap \overline{L_2} = \overline{L_2}$.

Since **CFLs** aren't closed under complementation, this means $L_1 - L_2$ **NEED NOT** be a CFL!

Hence**, False.**

(D) $L_1 \cup L_2 \rightarrow$ **CFL**

Since **CFLs** are **closed under union operation** and a regular language is also a CFL.

Hence, **True**.

Ref: Closure Property of Language Families

References

2 votes -- Hira (13.8k points)

5.2.26 Context Free Languages: GATE CSE 2021 Set 2 | Question: 41 top https://gateoverflow.in/357499

✔ In all the options, $w, x \in \{0,1\}^*$.

Option B :

ww^Rxx^R is CFL because ww^R is CFL and xx^R is CFL and we know concatenation of two CFL is CFL.

CFG for ww^Rxx^R :

$S \rightarrow AB$

$A \rightarrow 0A0 \mid 1A1 \mid \epsilon$; (A will produce ww^R)

$B \rightarrow 0B0 \mid 1B1 \mid \epsilon$; (B will produce xx^R)

We can also write CFG in simple manner as following :

$S \rightarrow AA$ (NOTE that both A will produce palindrome strings which may not be same)

$A \rightarrow 0A0 \mid 1A1 \mid \epsilon$; (A will produce ww^R)

Option C :

wxw^R is Σ^* because every string u can be written as wxw^R by taking $w = \epsilon$ and $u = x$

Every regular language is CFL, so, Option C is CFL.

Option D :

wxx^Rw^R is basically $uu^R; u \in \{0,1\}^*$.

Claim 1 :

Every string of the form wxx^Rw^R is basically even length palindrome :

Proof :

Let $w = a_1a_2a_3; x = b_1b_2$ then $wxx^Rw^R = a_1a_2a_3b_1b_2b_2b_1a_3a_2a_1$ is even length palindrome uu^R where $u = a_1a_2a_3b_1b_2$

Claim 2 :

Every even length palindrome uu^R can be written as wxx^Rw^R.

Proof :

Take $x = \in$. That's it.

So,

wxx^Rw^R is basically $uu^R; u \in \{0,1\}^*$ which is well known CFL.

CFG for $uu^R; u \in \{0,1\}^*$:

$A \rightarrow 0A0 \mid 1A1 \mid \in$; (A will produce uu^R)

Option A :

$L = wxw^Rx^R$ is Not CFL. We can prove it by using pumping lemma for CFLs.

Assume L is CFL then we have pumping length P.

Assume P is the pumping length for L. Take string $W = 0^P1^P0^P1^P$ in L. Since $|u| \geq P$, so we should be able to pump it somehow. But whatever correct decomposition/split $uvxyz$ of W we take, when we do raise v, y to the power zero, the resulting string will no longer belong to the language L. So we have contradiction, hence, L is Not CFL.

This is correct formal argument for L to be Non CFL and more examples to understand "Pumping lemma of CFL or regular languages" can be found in my answers in my profile. So, anyone who want to learn, can practice from there.

Informal and vague Idea of $L = wxw^Rx^R$ being Non-CFL is as follows :

We need to store w in stack so that we can compare it with w^R in future. So, we push w on stack. We need to store x in stack so that we can compare it with x^R in future. So, we push x on stack. So, after scanning substring/prefix wx , our stack has x on top of w in the stack. Now, we need to match w^R with w but on top of the stack we have x, so we cannot match w^R with w. So, PDA can not accept this language.

1 votes -- Deepak Poonia (23.3k points)

5.2.27 Context Free Languages: GATE IT 2006 | Question: 34 top https://gateoverflow.in/3573

$A \rightarrow bA \mid \varepsilon$

$\therefore \quad A = b^*$

$S \rightarrow aSAb \mid \varepsilon$

$\equiv S \rightarrow aSb^*b \mid \varepsilon$

$\equiv S \rightarrow aSb^+ \mid \varepsilon$

$S = a^n(b^+)^n, \quad n \geq 0$

$S = a^nb^nb^*, \quad n \geq 0$

$S = a^mb^n, \quad m \leq n$

Hence, option B is correct.

62 votes -- Pragy Agarwal (18.3k points)

5.2.28 Context Free Languages: GATE IT 2006 | Question: 4 top https://gateoverflow.in/3543

$L(G) = PALINDROME$

$baba$ does not belong to palindrome , so **B** is the answer.

29 votes -- Praveen Saini (41.9k points)

5.2.29 Context Free Languages: GATE IT 2007 | Question: 46 top https://gateoverflow.in/3481

From Above grammar:

Regular expression for $G1$: $(x+z)^+ + (x+z)^*y(y+z)^+$

Regular expression for $G2$: $(y+z+xy)^+$

Option **A** is correct .

22 votes -- Praveen Saini (41.9k points)

5.2.30 Context Free Languages: GATE IT 2007 | Question: 48 top https://gateoverflow.in/3490

ii, iii and iv.

So, option **C** is correct.

Above grammar is for ***equal no of x and y***

from Non-terminal $S \rightarrow xB$

$\Rightarrow xy$ [as $B \rightarrow y$ one y for one x]

$S \rightarrow xB$

$\Rightarrow xxBB$ [as $B \rightarrow yBB$ one B result in one y for one x]

$S \rightarrow xB$

$\Rightarrow xyS$ [as $B \rightarrow yS$ one y for one x and start again]

Note: Same applies for string start with y i.e . $S \rightarrow yA$.

32 votes -- Praveen Saini (41.9k points)

5.2.31 Context Free Languages: GATE IT 2007 | Question: 49 top https://gateoverflow.in/3491

✔ Regular grammar is either right linear or left linear. A left linear grammar is one in which there is at most 1 non-terminal on the right side of any production, and it appears at the left most position. Similarly, in right linear grammar non-terminal appears at the right most position.

Here, we can write a right linear grammar for $G1$ as

$S \to w(E$
$E \to id)S$
$S \to o$

(w - WHILE, o - OTHER)

So, $L(G1)$ is regular.

Now for $G2$ also we can write a right linear grammar:

$S \to w(E$
$E \to id)S$
$E \to id + E$
$E \to id * E$
$S \to o$

making its language regular.
So, both $G1$ and $G2$ have an equivalent regular grammar. But given in the question both these grammars are neither right linear nor left linear and hence not a regular grammar. So, **D** must be the answer.

http://www.cs.odu.edu/~toida/nerzic/390teched/regular/grammar/reg-grammar.html

References

58 votes -- Arjun Suresh (330k points)

5.2.32 Context Free Languages: GATE IT 2008 | Question: 34 top https://gateoverflow.in/3344

✔ $B \to 0B00 \mid 1$

generates $\{0^n 10^{2n} \mid n \geq 0\}$

$S \to AA,$
$A \to 0A \mid A0 \mid 1$

generates $0A0A \to 00A0A \to 00101$, which is there in only B and D choices. D is not the answer as "00" is not generated by the given grammar. So, only option left is B and if we see carefully, non-terminal B is generating the second part of B choice and AA is generating the first part.

$\{0^i 10^j 10^k \mid i, j, k \geq 0\} \cup \{0^n 10^{2n} \mid n \geq 0\}$

35 votes -- Arjun Suresh (330k points)

5.2.33 Context Free Languages: GATE IT 2008 | Question: 78 top https://gateoverflow.in/3392

✔ $S \to aS$

$S \to aA$

$S \to aaAb$

$S \to aabAab$

$S \to aabbAaab$

$S \to aabbaab$

Hence, (**D**) is the answer.

39 votes -- Shreyans Dhankhar (2.1k points)

5.3 Countable Uncountable Set (3) top

5.3.1 Countable Uncountable Set: GATE CSE 1997 | Question: 3.4 top https://gateoverflow.in/2235

Given $\Sigma = \{a, b\}$, which one of the following sets is not countable?

A. Set of all strings over Σ
B. Set of all languages over Σ
C. Set of all regular languages over Σ
D. Set of all languages over Σ accepted by Turing machines

gate1997 theory-of-computation normal countable-uncountable-set

Answer

5.3.2 Countable Uncountable Set: GATE CSE 2014 Set 3 | Question: 16 top https://gateoverflow.in/2050

Let Σ be a finite non-empty alphabet and let 2^{Σ^*} be the power set of Σ^*. Which one of the following is **TRUE**?

A. Both 2^{Σ^*} and Σ^* are countable
B. 2^{Σ^*} is countable and Σ^* is uncountable
C. 2^{Σ^*} is uncountable and Σ^* is countable
D. Both 2^{Σ^*} and Σ^* are uncountable

gate2014-cse-set3 theory-of-computation normal countable-uncountable-set

Answer

5.3.3 Countable Uncountable Set: GATE CSE 2019 | Question: 34 top https://gateoverflow.in/302814

Consider the following sets:

S1: Set of all recursively enumerable languages over the alphabet $\{0, 1\}$

S2: Set of all syntactically valid C programs

S3: Set of all languages over the alphabet $\{0, 1\}$

S4: Set of all non-regular languages over the alphabet $\{0, 1\}$

Which of the above sets are uncountable?

A. S1 and S2
B. S3 and S4
C. S2 and S3
D. S1 and S4

gate2019-cse theory-of-computation countable-uncountable-set

Answer

Answers: Countable Uncountable Set

5.3.1 Countable Uncountable Set: GATE CSE 1997 | Question: 3.4 top https://gateoverflow.in/2235

✔ Correct Option: **B**

Set of all languages over Σ is uncountable.

Ref: http://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-045j-automata-computability-and-complexity-spring-2011/lecture-notes/MIT6_045JS11_lec05.pdf

---

Power set of an infinite set is uncountable. Set of languages over Σ is the power set of set of strings over Σ which is an infinite set. Hence the set of languages becomes an uncountable set.

References

31 votes -- Arjun Suresh (330k points)

5.3.2 Countable Uncountable Set: GATE CSE 2014 Set 3 | Question: 16 top https://gateoverflow.in/2050

✔ A set is countable means there exist a enumeration procedure to generate each of its elements and for a given element of set, it take finite step to generate it using the enumeration procedure.

Let $\Sigma = \{a, b\}$ and there exist a enumeration procedure to generate all the string of the language Σ^*.

$\Sigma^* = \{\epsilon, a, b, aa, ab, ba, bb, aaa, \ldots\}$

Here, enumeration procedure is simply the generating string of the language by length for the fixed length string are in alphabetical order.

This way Σ^* is countably infinite & 2^{Σ^*} will be uncountable set

Because the power set of any countably infinite set is uncountable.

Ref: http://www.cs.xu.edu/csci250/06s/Theorems/powerSetuncountable.pdf

Correct Answer: C

References

34 votes -- Sandeep Singh (5.8k points)

5.3.3 Countable Uncountable Set: GATE CSE 2019 | Question: 34 top https://gateoverflow.in/302814

✔ Every TM can be encoded with 0's and 1's, it means that every TM can be represented by a unique binary number.

Let $\Sigma = \{0, 1\}$ then set of all binary strings $= \Sigma^*$. We know that, Σ^* is countable $\Rightarrow$ Set of all TM's is countable.

S1: Set of all recursively enumerable languages over the alphabet $\{0, 1\}$

Every Recursively enumerable language have a TM, and set of all TM's is countable. Therefore, **set of all Recursively enumerable languages is countable**.

S2: Set of all syntactically valid C programs. We can make a one to one equivalence for all valid C programs and all valid TM encodings. Since, the set of all valid TM encodings is countable, it means that the **set of all syntactically valid C programs is also countable.**

Therefore **Set of all syntactically valid C programs are countable** .

S3: Set of all languages over the alphabet $\{0, 1\}$. It is 2^{Σ^*} which is uncountable as the power set of an infinite set is uncountable.

S4: Set of all non-regular languages over the alphabet $\{0, 1\}$

We know that, set of all languages is uncountable and set of all Regular Languages is countable. So, the set of all non-regular languages should be uncountable as union of two countable sets cannot make an uncountable set.

Answer is **B.**

40 votes -- Shaik Masthan (50.4k points)

5.4 Decidability (28) top

5.4.1 Decidability: GATE CSE 1987 | Question: 2l top https://gateoverflow.in/80603

State whether the following statement are TRUE or FALSE.

A is recursive if both A and its complement are accepted by Turing machines.

gate1987 theory-of-computation turing-machine decidability true-false

Answer

5.4.2 Decidability: GATE CSE 1987 | Question: 2m top ☛ https://gateoverflow.in/80606

State whether the following statements are TRUE or FALSE:

The problem as to whether a Turing machine M accepts input w is undecidable.

gate1987 theory-of-computation turing-machine decidability true-false

Answer

5.4.3 Decidability: GATE CSE 1988 | Question: 2viii top ☛ https://gateoverflow.in/93948

State the halting problem of the Turing machine.

gate1988 theory-of-computation descriptive decidability turing-machine

Answer

5.4.4 Decidability: GATE CSE 1989 | Question: 3-iii top ☛ https://gateoverflow.in/87123

Which of the following problems are undecidable?

A. Membership problem in context-free languages.
B. Whether a given context-free language is regular.
C. Whether a finite state automation halts on all inputs.
D. Membership problem for type 0 languages.

gate1989 normal theory-of-computation decidability multiple-selects

Answer

5.4.5 Decidability: GATE CSE 1990 | Question: 3-vii top ☛ https://gateoverflow.in/84835

It is undecidable whether:

A. An arbitrary Turing machine halts after 100 steps.
B. A Turing machine prints a specific letter.
C. A Turing machine computes the products of two numbers
D. None of the above.

gate1990 normal theory-of-computation decidability multiple-selects

Answer

5.4.6 Decidability: GATE CSE 1995 | Question: 11 top ☛ https://gateoverflow.in/2647

Let L be a language over Σ i.e., $L \subseteq \Sigma^*$. Suppose L satisfies the two conditions given below.

i. L is in NP and
ii. For every n, there is exactly one string of length n that belongs to L.

Let L^c be the complement of L over Σ^*. Show that L^c is also in NP.

gate1995 theory-of-computation normal decidability proof descriptive

Answer

5.4.7 Decidability: GATE CSE 1996 | Question: 1.9 top ☛ https://gateoverflow.in/2713

Which of the following statements is false?

A. The Halting Problem of Turing machines is undecidable
B. Determining whether a context-free grammar is ambiguous is undecidable
C. Given two arbitrary context-free grammars G_1 and G_2 it is undecidable whether $L(G_1) = L(G_2)$
D. Given two regular grammars G_1 and G_2 it is undecidable whether $L(G_1) = L(G_2)$

gate1996 theory-of-computation decidability easy

Answer

5.4.8 Decidability: GATE CSE 1997 | Question: 6.5 top https://gateoverflow.in/2261

Which one of the following is not decidable?

A. Given a Turing machine M, a string s and an integer k, M accepts s within k steps
B. Equivalence of two given Turing machines
C. Language accepted by a given finite state machine is not empty
D. Language generated by a context free grammar is non-empty

gate1997 theory-of-computation decidability easy

Answer

5.4.9 Decidability: GATE CSE 1999 | Question: 10 top https://gateoverflow.in/1509

Suppose we have a function HALTS which when applied to any arbitrary function f and its arguments will say TRUE if function f terminates for those arguments and FALSE otherwise. Example: Given the following function definition.

FACTORIAL (N) = IF (N=0) THEN 1 ELSE N*FACTORIAL (N-1)

Then HALTS (FACTORIAL, 4) = TRUE and HALTS (FACTORIAL, -5) = FALSE

Let us define the function FUNNY (f) = IF HALTS (f) THEN not (f) ELSE TRUE

a. Show that FUNNY terminates for all functions f.
b. use (a) to prove (by contradiction) that it is not possible to have a function like HALTS which for arbitrary functions and inputs says whether it will terminate on that input or not.

gate1999 theory-of-computation descriptive decidability

Answer

5.4.10 Decidability: GATE CSE 2000 | Question: 2.9 top https://gateoverflow.in/656

Consider the following decision problems:

$(P1)$: Does a given finite state machine accept a given string?

$(P2)$: Does a given context free grammar generate an infinite number of strings?

Which of the following statements is true?

A. Both$(P1)$ and $(P2)$ are decidable
B. Neither $(P1)$ nor $(P2)$ is decidable
C. Only $(P1)$ is decidable
D. Only $(P2)$ is decidable

gate2000-cse theory-of-computation decidability normal

Answer

5.4.11 Decidability: GATE CSE 2001 | Question: 2.7 top https://gateoverflow.in/725

Consider the following problem X.

Given a Turing machine M over the input alphabet Σ, any state q of M and a word $w \in \Sigma^*$, does the computation of M on w visit the state of q?

Which of the following statements about X is correct?

A. X is decidable
B. X is undecidable but partially decidable
C. X is undecidable and not even partially decidable
D. X is not a decision problem

gate2001-cse theory-of-computation decidability normal

Answer

5.4.12 Decidability: GATE CSE 2001 | Question: 7 top

https://gateoverflow.in/748

Let a decision problem X be defined as follows:

X: Given a Turing machine M over Σ and any word $w \in \Sigma$, does M loop forever on w?

You may assume that the halting problem of Turing machine is undecidable but partially decidable.

A. Show that X is undecidable
B. Show that X is not even partially decidable

gate2001-cse theory-of-computation decidability turing-machine easy descriptive

Answer

5.4.13 Decidability: GATE CSE 2002 | Question: 14 top

https://gateoverflow.in/867

The aim of the following question is to prove that the language $\{M \mid M$ is the code of the Turing Machine which, irrespective of the input, halts and outputs a $1\}$, is undecidable. This is to be done by reducing from the language $\{M', x \mid M'$ halts on $x\}$, which is known to be undecidable. In parts (a) and (b) describe the 2 main steps in the construction of M. In part (c) describe the key property which relates the behaviour of M on its input w to the behaviour of M' on x.

A. On input w, what is the first step that M must make?
B. On input w, based on the outcome of the first step, what is the second step M must make?
C. What key property relates the behaviour of M on w to the behaviour of M' on x?

gate2002-cse theory-of-computation decidability normal turing-machine descriptive difficult

Answer

5.4.14 Decidability: GATE CSE 2003 | Question: 52 top

https://gateoverflow.in/356

Consider two languages L_1 and L_2 each on the alphabet Σ. Let $f : \Sigma^* \to \Sigma^*$ be a polynomial time computable bijection such that $(\forall x)[x \in L_1 \text{ iff } f(x) \in L_2]$. Further, let f^{-1} be also polynomial time computable.

Which of the following **CANNOT** be true?

A. $L_1 \in P$ and L_2 is finite
B. $L_1 \in NP$ and $L_2 \in P$
C. L_1 is undecidable and L_2 is decidable
D. L_1 is recursively enumerable and L_2 is recursive

gate2003-cse theory-of-computation normal decidability

Answer

5.4.15 Decidability: GATE CSE 2005 | Question: 45 top

https://gateoverflow.in/1375

Consider three decision problems P_1, P_2 and P_3. It is known that P_1 is decidable and P_2 is undecidable. Which one of the following is TRUE?

A. P_3 is decidable if P_1 is reducible to P_3
B. P_3 is undecidable if P_3 is reducible to P_2
C. P_3 is undecidable if P_2 is reducible to P_3
D. P_3 is decidable if P_3 is reducible to P_2's complement

gate2005-cse theory-of-computation decidability normal

Answer

5.4.16 Decidability: GATE CSE 2007 | Question: 6 top

https://gateoverflow.in/1204

Which of the following problems is undecidable?

A. Membership problem for CFGs
B. Ambiguity problem for CFGs
C. Finiteness problem for FSAs
D. Equivalence problem for FSAs

gate2007-cse theory-of-computation decidability normal

Answer

5.4.17 Decidability: GATE CSE 2008 | Question: 10 top

https://gateoverflow.in/408

Which of the following are decidable?

I. Whether the intersection of two regular languages is infinite
II. Whether a given context-free language is regular
III. Whether two push-down automata accept the same language
IV. Whether a given grammar is context-free

A. I and II
B. I and IV
C. II and III
D. II and IV

gate2008-cse theory-of-computation decidability easy

Answer

5.4.18 Decidability: GATE CSE 2012 | Question: 24 top

https://gateoverflow.in/1608

Which of the following problems are decidable?

1. Does a given program ever produce an output?
2. If L is a context-free language, then, is $\bar{L}$ also context-free?
3. If L is a regular language, then, is $\bar{L}$ also regular?
4. If L is a recursive language, then, is $\bar{L}$ also recursive?

A. $1, 2, 3, 4$
B. $1, 2$
C. $2, 3, 4$
D. $3, 4$

gate2012-cse theory-of-computation decidability normal

Answer

5.4.19 Decidability: GATE CSE 2013 | Question: 41 top

https://gateoverflow.in/1553

Which of the following is/are undecidable?

1. G is a CFG. Is $L(G) = \phi$?
2. G is a CFG. Is $L(G) = \Sigma^*$?
3. M is a Turing machine. Is $L(M)$ regular?
4. A is a DFA and N is an NFA. Is $L(A) = L(N)$?

A. 3 only
B. 3 and 4 only
C. 1, 2 and 3 only
D. 2 and 3 only

gate2013-cse theory-of-computation decidability normal

Answer

5.4.20 Decidability: GATE CSE 2014 Set 3 | Question: 35 top https://gateoverflow.in/2069

Which one of the following problems is undecidable?

A. Deciding if a given context-free grammar is ambiguous.
B. Deciding if a given string is generated by a given context-free grammar.
C. Deciding if the language generated by a given context-free grammar is empty.
D. Deciding if the language generated by a given context-free grammar is finite.

gate2014-cse-set3 theory-of-computation context-free-languages decidability normal

Answer

5.4.21 Decidability: GATE CSE 2015 Set 2 | Question: 21 top https://gateoverflow.in/8111

Consider the following statements.

I. The complement of every Turing decidable language is Turing decidable
II. There exists some language which is in NP but is not Turing decidable
III. If L is a language in NP, L is Turing decidable

Which of the above statements is/are true?

A. Only II
B. Only III
C. Only I and II
D. Only I and III

gate2015-cse-set2 theory-of-computation decidability easy

Answer

5.4.22 Decidability: GATE CSE 2015 Set 3 | Question: 53 top https://gateoverflow.in/8562

Language L_1 is polynomial time reducible to language L_2. Language L_3 is polynomial time reducible to language L_2, which in turn polynomial time reducible to language L_4. Which of the following is/are true?

I. if $L_4 \in P$, then $L_2 \in P$
II. if $L_1 \in P$ or $L_3 \in P$, then $L_2 \in P$
III. $L_1 \in P$, if and only if $L_3 \in P$
IV. if $L_4 \in P$, then $L_3 \in P$

A. II only
B. III only
C. I and IV only
D. I only

gate2015-cse-set3 theory-of-computation decidability normal

Answer

5.4.23 Decidability: GATE CSE 2016 Set 1 | Question: 17 top https://gateoverflow.in/39651

Which of the following decision problems are undecidable?

I. Given NFAs N_1 and N_2, is $L(N_1) \cap L(N_2) = \Phi$
II. Given a CFG $G = (N, \Sigma, P, S)$ and a string $x \in \Sigma^*$, does $x \in L(G)\}$?
III. Given CFGs G_1 and G_2, is $L(G_1) = L(G_2)$?
IV. Given a TM M, is $L(M) = \Phi$?

A. I and IV only
B. II and III only
C. III and IV only
D. II and IV only

gate2016-cse-set1 theory-of-computation decidability easy

Answer

5.4.24 Decidability: GATE CSE 2017 Set 1 | Question: 39 top

https://gateoverflow.in/118322

Let A and B be finite alphabets and let $\#$ be a symbol outside both A and B. Let f be a total function from A^* to B^*. We say f is *computable* if there exists a Turing machine M which given an input $x \in A^*$, always halts with $f(x)$ on its tape. Let L_f denote the language $\left\{x\#f(x) \mid x \in A^*\right\}$. Which of the following statements is true:

A. f is computable if and only if L_f is recursive.
B. f is computable if and only if L_f is recursively enumerable.
C. If f is computable then L_f is recursive, but not conversely.
D. If f is computable then L_f is recursively enumerable, but not conversely.

gate2017-cse-set1 theory-of-computation decidability difficult

Answer

5.4.25 Decidability: GATE CSE 2017 Set 2 | Question: 41 top

https://gateoverflow.in/118605

Let $L(R)$ be the language represented by regular expression R. Let $L(G)$ be the language generated by a context free grammar G. Let $L(M)$ be the language accepted by a Turing machine M. Which of the following decision problems are undecidable?

I. Given a regular expression R and a string w, is $w \in L(R)$?
II. Given a context-free grammar G, is $L(G) = \emptyset$
III. Given a context-free grammar G, is $L(G) = \Sigma^*$ for some alphabet Σ?
IV. Given a Turing machine M and a string w, is $w \in L(M)$?

A. I and IV only
B. II and III only
C. II, III and IV only
D. III and IV only

gate2017-cse-set2 theory-of-computation decidability

Answer

5.4.26 Decidability: GATE CSE 2018 | Question: 36 top

https://gateoverflow.in/204110

Consider the following problems. $L(G)$ denotes the language generated by a grammar G. L(M) denotes the language accepted by a machine M.

I. For an unrestricted grammar G and a string w, whether $w \in L(G)$
II. Given a Turing machine M, whether $L(M)$ is regular
III. Given two grammarG_1 and G_2, whether $L(G_1) = L(G_2)$
IV. Given an NFA N, whether there is a deterministic PDA P such that N and P accept the same language

Which one of the following statement is correct?

A. Only I and II are undecidable
B. Only II is undecidable
C. Only II and IV are undecidable
D. Only I, II and III are undecidable

gate2018-cse theory-of-computation decidability easy

Answer

5.4.27 Decidability: GATE CSE 2020 | Question: 26 top

https://gateoverflow.in/333205

Which of the following languages are undecidable? Note that $\langle M \rangle$ indicates encoding of the Turing machine M.

- $L_1 = \{\langle M \rangle \mid L(M) = \varnothing\}$
- $L_2 = \{\langle M, w, q \rangle \mid M \text{ on input } w \text{ reaches state } q \text{ in exactly } 100 \text{ steps}\}$
- $L_3 = \{\langle M \rangle \mid L(M) \text{ is not recursive}\}$

- $L_4 = \{\langle M \rangle \mid L(M) \text{contains at least 21 members}\}$

A. L_1, L_3, and L_4 only
B. L_1 and L_3 only
C. L_2 and L_3 only
D. L_2, L_3, and L_4 only

gate2020-cse theory-of-computation decidability

Answer

5.4.28 Decidability: GATE CSE 2021 Set 2 | Question: 36 top

☛ https://gateoverflow.in/357504

Consider the following two statements about regular languages:

- S_1: Every infinite regular language contains an undecidable language as a subset.
- S_2: Every finite language is regular.

Which one of the following choices is correct?

A. Only S_1 is true
B. Only S_2 is true
C. Both S_1 and S_2 are true
D. Neither S_1 nor S_2 is true

gate2021-cse-set2 theory-of-computation regular-languages decidability

Answer

Answers: Decidability

5.4.1 Decidability: GATE CSE 1987 | Question: 2l top

☛ https://gateoverflow.in/80603

✔ Yes if A and its complement are accepted by Turing machines.then A is recursive.

Suppose a language A is recursively enumerable. That means there exists a Turing machine $T1$ that, given any string of the language, halts and accepts that string.

Now let's also suppose that the complement of A, $A' = \{w : w \mid A\}$, is recursively enumerable. That means there is some other Turing machine $T2$ that, given any string of A' halts and accepts that string. So any string belongs to either A or A'. Hence, any string will cause either $T1$ or $T2$ (or both) to halt. We construct a new Turing machine that emulates both $T1$ and $T2$, alternating moves between them. When either one stops, we can tell (by whether it accepted or rejected the string) to which language the string belongs.

Thus, we have constructed a Turing machine that, for each input, halts with an answer whether or not the string belongs to A'. Therefore A and A' are recursive languages.

24 votes -- Prashant Singh (47.1k points)

5.4.2 Decidability: GATE CSE 1987 | Question: 2m top

☛ https://gateoverflow.in/80606

✔ Yes. The problem as to whether a Turing machine M accepts input w is undecidable which is the well known Halting Problem.

If string w is going to loop then we can not determine if it will be eventually accepted by TM or not.

23 votes -- Prashant Singh (47.1k points)

5.4.3 Decidability: GATE CSE 1988 | Question: 2viii top

☛ https://gateoverflow.in/93948

✔ Halting Problem: Given an input Turing machine M (in some encoded form) and a word w, we have to decide if M halts on w.

In programming terms, this is equivalent to saying if a program will finish running on a given input.

Now, answering this problem is straightforward if the answer is "yes" -- we just have to run the TM on the given input and say "yes" when it halts. But answering "no" is not straight forwards as the TM may go to an infinite loop from which we are no sure if it'll eventually come out or not.

Thus Halting problem albeit semi-decidable is undecidable. It's complement is not even semi-decidable.

9 votes -- Arjun Suresh (330k points)

5.4.4 Decidability: GATE CSE 1989 | Question: 3-iii top https://gateoverflow.in/87123

✔
1. Membership problem in context-free languages. is **Decidable.**
2. Whether a given context-free language is regular. **Undecidable [Regularity is decidable till DCFL class]**
3. Whether a finite state automation halts on all inputs. **Decidable**
4. Membership problem for type 0 languages. **Undecidable [undecidable for RE or semi-decidable]**

Ref : http://gatecse.in/grammar-decidable-and-undecidable-problems/

References

28 votes -- Prashant Singh (47.1k points)

5.4.5 Decidability: GATE CSE 1990 | Question: 3-vii top https://gateoverflow.in/84835

✔
A. An arbitrary Turing machine halts after 100 steps. **DECIDABLE,**we can run TM for 100 steps and conclude that. (assuming "after 100 steps" mean "exactly after 100 steps")
B. A Turing machine prints a specific letter. **UNDECIDABLE,**
C. A Turing machine computes the products of two numbers, **UNDECIDABLE,** Even though we can desing a TM for calculation product of 2 numbers but here it is asking whether given TM computes product of 2 numbers,so the behaviour of TM unknown hence,Undecidable.

51 votes -- Prajwal Bhat (7.6k points)

5.4.6 Decidability: GATE CSE 1995 | Question: 11 top https://gateoverflow.in/2647

✔ Since L is in NP it is decidable (recursive) and so is its complement L^c. Now, L^c may or may not be in NP. But we are given that for any string length n, exactly one string belong to L, which means for any string length all but one string belong to L^c.

Now, definition of NP says that all "yes" instances of the problem can be solved in polynomial time using a nondeterministic TM. So, given an instance of $\langle L^c, x \rangle$, we non-deterministically take all words of length n, where n is the length of w, and see if it is in L. As soon as we get the word (such a word is sure to exist as exactly one word of length n belong to L), we see if this word is same as x. If it is not same (and only if it is not same), $x \in L^c$ and we get this answer in polynomial time making L^c an NP problem.

References

22 votes -- Arjun Suresh (330k points)

5.4.7 Decidability: GATE CSE 1996 | Question: 1.9 top https://gateoverflow.in/2713

✔ Answer is **D.**

Equivalence of Regular languages is decidable.

1. Membership,
2. Emptiness,
3. Finiteness,
4. Equivalence,
5. Ambiguity,
6. Regularity,

7. Everything,
8. Disjointedness...

All are decidable for Regular languages.

First 3 for CFL.

Only 1^{st} for CSL and REC.

None for RE.

30 votes -- Gate Keeda (15.9k points)

5.4.8 Decidability: GATE CSE 1997 | Question: 6.5 top https://gateoverflow.in/2261

✔ **Option B.** Equivalence of two TMs is undecidable.

Option (A) is not halting problem. In Halting problem no. of steps can go up to infinity and that is the only reason why it becomes undecidable. In **(A)** the number of steps is restricted to a finite number 'k' and simulating a TM for 'k' steps is trivially decidable because we just go to step k and output the answer.

For **Options (C)** and **(D)** we do have well defined algorithms making them decidable.

36 votes -- Bhagirathi Nayak (11.7k points)

5.4.9 Decidability: GATE CSE 1999 | Question: 10 top https://gateoverflow.in/1509

(a). FUNNY(f) takes as input a function f. Now we have a function HALTS which tells us whether f halts or not. i.e., if HALTS returns TRUE f halts and if HALTS returns FALSE f won't halt.

So, in FUNNY, we can call HALT(f) and if the return value is

1. TRUE, then we call function f and return its negated value.
2. FALSE, then we just return TRUE

Since, HALT function exist and is guaranteed to return for any input function and since we are calling f only when it is known to terminate, FUNNY(f) will terminate for any input function f.

(b). We can use FUNNY(f) to determine is a function f does not terminate (non-halting problem) which is a known undecidable (not even semi-decidable) problem.

1 votes -- Arjun Suresh (330k points)

5.4.10 Decidability: GATE CSE 2000 | Question: 2.9 top https://gateoverflow.in/656

✔ For $P1$, we just need to give a run on the machine. Finite state machines always halts unlike TM.

For$P2$, check if the CFG generates any string of length between n and $2n-1$, where n is the pumping lemma constant. If So, $L(CFG)$ is infinite, else finite. Finding the pumping lemma constant is not trivial - but there are other procedures which can do this - http://cs.stackexchange.com/questions/52507/is-it-decidable-whether-a-given-context-free-grammar-generates-an-infinite-numbe/52520

Hence, both $P1$ and $P2$ are decidable - answer is (**A**).

http://gatecse.in/wiki/Grammar:_Decidable_and_Undecidable_Problems

References

32 votes -- Arjun Suresh (330k points)

5.4.11 Decidability: GATE CSE 2001 | Question: 2.7 top https://gateoverflow.in/725

✔ X is undecidable but partially decidable.

We have the TM M. Just make the state q the final state and make all other final states non-final and get a new TM M'.

Give input w to M'. If w would have taken M to state q (yes case of the problem), our new TM M' would accept it. So, the given problem is partially decidable.

If M goes for an infinite loop and never reaches state q (no case for the problem), M' cannot output anything. This problem is the state entry problem, which like word accepting problem and halting problem is undecidable.

65 votes -- Arjun Suresh (330k points)

5.4.12 Decidability: GATE CSE 2001 | Question: 7 top

https://gateoverflow.in/748

✔ The question asks if M loop forever on w. If M loop forever on w, M wouldn't halt on w. And if M doesn't halt on w, M should loop forever. So, this problem is exactly same as asking if " M doesn't halt on w", which is the complement of halting problem and is not even partially decidable. So, X is not even partially decidable.

38 votes -- Arjun Suresh (330k points)

5.4.13 Decidability: GATE CSE 2002 | Question: 14 top

https://gateoverflow.in/867

✔

a. M erases its input w and simulate the moves of M' on x. Thus if M' halts on x, M accepts any input (Σ^*) and if M' doesn't halt on x, M accepts no string (ϕ)

b. Give the description of M - <M> to the TM that decides L. If TM accepts <M>, M halts on all inputs $\rightarrow M'$ accept x. If TM rejects <M>, M doesn't halt on some input $\rightarrow M'$ doesn't halt on x, due to our construction of M in 1st step. Thus we decide halting problem

c. M halting on all inputs w is the key property relating to M' which is halting on a given input x

11 votes -- Arjun Suresh (330k points)

5.4.14 Decidability: GATE CSE 2003 | Question: 52 top

https://gateoverflow.in/356

✔ Since, f is a polynomial time computable bijection and f^{-1} is also polynomial time computable, L_1 and L_2 should have the same complexity (isomorphic). This is because, given a problem for L_1, we can always do a polynomial time reduction to L_2 and vice verse. Hence, the answer is 'C', as in 'A', L_1 and L_2 can be finite, in 'B', L_1 and L_2 can be in P and in 'D', L_1 and L_2 can be recursive. Only, in '**C**' there is no intersection for L_1 and L_2, and hence it can't be true.

Alternatively, we can prove 'C' to be false as follows:
Given L_2 is decidable. Now, for a problem in L_1, we can have a TM, which takes an input x, calculates $f(x)$ in polynomial time, check $f(x)$ is in L_2 (this is decidable as L_2 is decidable), and if it is, then output yes and otherwise no. Thus L_1 must also be decidable.

35 votes -- gatecse (62.6k points)

5.4.15 Decidability: GATE CSE 2005 | Question: 45 top

https://gateoverflow.in/1375

✔

A. If P_1 is reducible to P_3, then P_3 is at least as hard as P_1. So, no guarantee if P_3 is decidable.
B. If P_3 is reducible to P_2, then P_3 cannot be harder than P_2. But P_2 being undecidable, this can't say P_3 is undecidable.
C. If P_2 is reducible to P_3, then P_3 is at least as hard as P_2. Since, P_2 is undecidable, this means P_3 is also undecidable **-hence the answer**.
D. Complement of an undecidable problem is undecidable. Now, reducing to an undecidable problem can't prove P_3 is decidable.

http://gatecse.in/wiki/Some_Reduction_Inferences

References

52 votes -- Arjun Suresh (330k points)

5.4.16 Decidability: GATE CSE 2007 | Question: 6 top

https://gateoverflow.in/1204

✔

- Membership problem is decidable as it can be solved by parsers.
- Finiteness problem is decidable for FSAs (also for CFGs), as we just need to check for a loop in the DFA.
- Equivalence problem for FSAs is decidable as we can take the complement of one FSA (complement of FSA is another FSA), and do an intersection with the other (FSAs are closed under intersection also), giving a new FSA. If this new FSA accept no string, then the given FSAs are equivalent, else not equivalent.
- Only ambiguity problem for CFGs are undecidable.

http://gatecse.in/wiki/Grammar:_Decidable_and_Undecidable_Problems

Correct Answer: B

References

32 votes -- Arjun Suresh (330k points)

5.4.17 Decidability: GATE CSE 2008 | Question: 10 top

https://gateoverflow.in/408

✔ Lets see options one by one :

I. The language here will be regular as intersection of regular languages will lead to regular language only. And we know that given a regular language, whether it is finite or not is a decidable problem. This can be seen by observing the DFA -- if DFA contains a state which contains a loop and that state is reachable from the start state and that state is either a final state or leading to final state, then the language will be infinite.

II. The regularity property is undecidable for context free languages. Hence, it is undecidable. Details regarding this : https://cs.stackexchange.com/questions/19482/why-is-deciding-regularity-of-a-context-free-language-undecidable

III. Now equivalence of two CFLs is also an undecidable property. Hence, given 2 PDAs which is nothing but characterizing CFLs, whether the 2 CFLs will be same or not cannot be decided.

IV. Given a grammar, it is context free iff its productions are of the type $V \to (V \cup \Sigma)^*$ which can be verified with a Turing machine. Hence, it is a decidable property..

Hence, (B) should be correct answer.

References

37 votes -- HABIB MOHAMMAD KHAN (67.5k points)

5.4.18 Decidability: GATE CSE 2012 | Question: 24 top

https://gateoverflow.in/1608

CFL's are not closed under complementation and a program can loop forever. So, it may not produce any output.

Regular and recursive languages are closed under complementation.

Hence, only **3,4** are decidable.

Correct Answer: D

34 votes -- Bhagirathi Nayak (11.7k points)

5.4.19 Decidability: GATE CSE 2013 | Question: 41 top

https://gateoverflow.in/1553

✔ Correct Option: **D**

1. First is Emptiness for CFG.
2. Second is everything for CFG.

3. Third is Regularity for REC
4. Fourth is equivalence for regular.

👍 32 votes -- Gate Keeda (15.9k points)

5.4.20 Decidability: GATE CSE 2014 Set 3 | Question: 35 top https://gateoverflow.in/2069

✔ (**A**) is the answer. Proving (A) is undecidable is not so easy. But we can easily prove the other three options given here are decidable.

https://gatecse.in/grammar-decidable-and-undecidable-problems/

References

👍 27 votes -- Arjun Suresh (330k points)

5.4.21 Decidability: GATE CSE 2015 Set 2 | Question: 21 top https://gateoverflow.in/8111

✔

I. is true. The solution to a decision problem is either "yes" or "no", and hence if we can decide a problem, we have also decided its complement- just reverse "yes" and "no". (This is applicable for decidability and not for acceptance)

II. is false. Because NP class is defined as the class of languages that can be solved in polynomial time by a non-deterministic Turing machine. So, none of the NP class problems is undecidable.

III. is true for same reason as II.

So, answer is **D**.

👍 55 votes -- Arjun Suresh (330k points)

5.4.22 Decidability: GATE CSE 2015 Set 3 | Question: 53 top https://gateoverflow.in/8562

✔

1. L_1 is polynomial time reducible to L_2. So, L_2 is at least as hard as L_1.
2. L_3 is polynomial time reducible to L_2. So, L_2 is at least as hard as L_3.
3. L_2 is polynomial time reducible to L_4. So, L_4 is at least as hard as L_2.

If L_4 is in P, L_3, L_2 and L_1 must also be in P. So, I and IV are true.

We can have L_1 in P and L_2 not in P, and none of the given conditions are violated. So, II is false.

Assume L_3 not in P. Now, Since L_2 must be at least as hard as L_3, it must also be not in P. But L_1 is less harder than L_1 as per condition 1, and it can be in P without violating any given conditions. So, III is false.

Hence, **C** is choice.

More Info: http://gatecse.in/wiki/Some_Reduction_Inferences

References

👍 43 votes -- Arjun Suresh (330k points)

5.4.23 Decidability: GATE CSE 2016 Set 1 | Question: 17 top https://gateoverflow.in/39651

✔

I. is Decidable, we may use cross product of NFA (or by converting them into DFA) , if We didn't get final states of both together at any state in it. then $L(N_1) \cap L(N_2) = \phi$, Disjoint languages.
II. Membership in CFG is Decidable (CYK algorithm)
III. Equivalence of Two context free grammars is Undecidable.
IV. For TM M , $L(M) = \phi$ is Undecidable.

Correct Answer: C

57 votes -- Praveen Saini (41.9k points)

5.4.24 Decidability: GATE CSE 2017 Set 1 | Question: 39 top

https://gateoverflow.in/118322

f being computable, given x, we get $f(x)$.

L_f is recursive - given any string we get to know if the string is in L or not.

Now lets see the two cases:

- Does f being computable implies L_f is recursive?

We are given x and we need to know if $x\#f(x)$ belongs to L_f. Since f being computable we can calculate $f(x)$ and our problem reduces to 3 string comparisons - first x, followed by # and then f(x) which can be done by a TM. So, L_f must be recursive.

- Does L_f being recursive implies f be computable?

If L_f is recursive, given any string we can say whether it belongs to L_f or not. Now, to compute $f(x)$, we can do a dovetailing approach starting with strings in lexicographic (or any other order) as follows $s1, s2, s3, \ldots$ and forming inputs to the TM as

$x\#s_1, x\#s_2, x\#s_3, \ldots$

By dovetailing it means to simulation TM for one step in first iteration, 2 steps in second iteration, 3 steps in third iteration etc., and in each iteration adding a new string to the simulation (TM is parallelly simulating all strings).

https://courses.engr.illinois.edu/cs373/sp2009/lectures/lect_24.pdf

Now, for any x, we should have an $f(x)$ since f is a total function. So, in our simulation we are sure that sometime we will add the string $x\#f(x)$ to the simulation and then the TM will halt for it. And whenever it does halt, we can just take the part of the input string after #, and that is $f(x)$. We have a way to compute $f(x)$ for any x. So, f is computable. The same method would work even if L_f is recursively enumerable and not necessarily recursive.

So, the correct statements are

1. If L_f is recursively enumerable f is computable
2. If f is computable L_f is recursive

So, option B is TRUE but official key was only A.

Option A is false because f is computable even when L_f is recursively enumerable and not necessarily recursive.

https://cs.stackexchange.com/questions/70626/recursive-language-and-computable-function

References

61 votes -- Arjun Suresh (330k points)

5.4.25 Decidability: GATE CSE 2017 Set 2 | Question: 41 top

https://gateoverflow.in/118605

✔ 1^{st} statement is Membership problem of regular language = **decidable**

2^{nd} statement is Emptyness problem of CFL = **decidable**

3^{rd} statement is accept everthing problem of CFL= un**decidable**

4^{th} statement is Membership problem of RE language = un**decidable**

D is answer.

29 votes -- Prashant Singh (47.1k points)

5.4.26 Decidability: GATE CSE 2018 | Question: 36 top

https://gateoverflow.in/204110

✔ 4th Statement : **Given an NFA N, whether there is a deterministic PDA P such that N and P accept the same language**

Is **Decidable** because We can Always say that There will definitely be a DPDA (and for that matter PDA too) which will accept the same language that NFA N is accepting. **But Careful,** Saying that (From other answers for this question) "PDA (accepting CFL) having more power than NFA (accepting regular) so we can decide whether both will accept the same language or not." is **WRONG.**

"Given a <PDA> P and a NFA N, Deciding Whether they both accept the same language or not" is Undecidable. "

"Given a <DPDA> D and a NFA N, Deciding Whether they both accept the same language or not" is Decidable. "

---

*3rd Option (***Third Statement** *)* **: Given two grammars G1 and G2, whether L(G1) = L(G2)**

is **Undecidable**. Because When nothing is mentioned about the type of the Grammar, It, by default, should be taken as **A Valid Grammar i.e. Type 0 Grammar which itself covers All the Grammars.**

So, Now the given problem is nothing but " **Equivalence of two RE Grammars i.e. Equality of Two RE languages** " Problem. **Which is Undecidable.**

Many students are confusing this statement with that of "Propositional Logic" statements. Which is not the case here. Let me elaborate : We all know that Equivalence of RE languages is Undecidable...But One could argue that Some RE languages are Regular also and for Regular languages, Equivalence Problem is Decidable. So Saying that "Equivalence of RE languages is Undecidable" would seem wrong. But We Know that It is NOT.

And This is because When we say "Decidable", It means that there is an Algorithm (Automation) to solve that problem and If that Problem is really decidable then You should be able to give an Algorithm for that, which for **All Valid Instances** should **Halt and Say Yes/No.**

"Equality" and "Equivalence" are two different things. **Equality of Grammars (Type 0-3 Grammars) is Decidable**, But **Equivalence is NOT** Because **Equivalence of Two Grammars is a relation defined by "Equality of Their Corresponding Languages" , Which is Undecidable for Type-0 Grammars.**

From the Comments on the question "In order to prove a statement wrong, we need only one counter example.Statement was " L1 = L2 is undecidable". It should not be true as it is decidable in case of regular languages."" ...

See, "In order to prove a statement wrong, we need only one counter example" **is a Generalized statement (NOT A THEOREM or A Proven Fact)** usually used in Logic.. But **"Generalization" is the Enemy/opposite of "Specification / particularization / Specific ".**

Correct Answer: D

63 votes -- Deepak Poonia (23.3k points)

5.4.27 Decidability: GATE CSE 2020 | Question: 26 top https://gateoverflow.in/333205

✔ We can answer this question just using Rice's theorem which states as follows:

> Any non-trivial property of the LANGUAGE recognizable by a Turing machine (recursively enumerable language) is undecidable

Trivial property of a set: For all instances of the set the property evaluates to True or for all instances of the set the property evaluates to False. i.e., without inspecting the "given instance" we can say whether it has the property or not. For example, "Language accepted by a TM is recursively enumerable". This is always true as any language accepted by a TM is called recursively enumerable by definition (it can also be recursive or context-sensitive or context-free or regular or finite also).

Non-trivial property of a set: For some instances of the set the property evaluates to True and for some it evaluates to False. For example,: "Language accepted by a TM is context free language". This is true if the language can also be accepted by some PDA but is false if no such PDA exists like for $L = \{ww \mid w \in \{0,1\}^*\}$.

$L_1 = \{\langle M \rangle \mid L(M) = \varnothing\}$

- L_1 is emptiness problem and is non-trivial because we can have two Turing machines M_1 and M_2 with $L(M_1) = \varnothing$ and $L(M_2) = \{0\}$ (something non-empty).
- So, L_1 is undecidable.

$L_3 = \{\langle M \rangle \mid L(M) \text{ is not recursive}\}$

- L_3 is also describing a non-trivial property (of the language of Turing machines) as the language of not all the Turing machines is recursive. For example, we can have a Turing machine for halting problem and its language is recursively enumerable but not recursive.

$L_4 = \{\langle M \rangle \mid L(M) \text{contains at least 21 members}\}$

- L_4 is also describing a non-trivial property as we can have two Turing machines M_1 and M_2 with say $L(M_1) = \{0\}$ and $L(M_2) = \{0^{2n} \mid n \geq 0\}$, where the property holds for $L(M_2)$ but not for $L(M_1)$. So, L_4 is also undecidable.

$L_2 = \{\langle M, w, q\rangle \mid M$ on input w reaches state q in exactly 100 steps$\}$

- This is actually a property of Turing machine and not its language. Obviously this is a non-trivial property (but of TM and not its language and hence Rice's theorem is not applicable). Here, we have to check if the given TM on given input w reaches state q in exactly 100 steps – certainly decidable as we just need to monitor the working of Turing machine for 100 steps which should happen in a finite amount of time. But if instead of 100 steps the question is modified to reaching state q ever, then the problem becomes state reachability problem and there is no guarantee that we can answer this problem in finite amount of time (we can answer "yes" but not necessarily "no"), and the problem becomes undecidable (but is still semi-decidable).

So A is correct.

https://gatecse.in/rices-theorem/

References

1 votes -- Arjun Suresh (330k points)

5.4.28 Decidability: GATE CSE 2021 Set 2 | Question: 36 top https://gateoverflow.in/357504

✔ S_1 : Every infinite regular language contains an undecidable language as a subset.

Cantor's theorem says that If S is **any** set then $|S| < |P(S)|$, where $P(S)$ is the power set of S.

So, from this we know that If we have any infinite set S then $P(S)$ is definitely Uncountable (Because remember that a set A is countable if and only if $|A| \leq |\mathbb{N}|$, where $\mathbb{N}$ is the set of natural numbers).
From this we can say that if S is countably infinite set, then $P(S)$ is uncountable.

We know that Every language is countable. (Every language L is a subset of Σ^*, and Σ^* is countable and subset of countable set is countable)
If we have any infinite language L, then it means that we have uncountably many subsets of L. But we know that set of all RE languages is countable. So, due to this we have a subset of L which is Not RE. So, the following statements are true :

1. **Every infinite regular language L has a subset S which is undecidable.**
2. **Every infinite regular language L has a subset S which is unrecognizable.**
3. **Every infinite language L has a subset S which is undecidable.**
4. **Every infinite language L has a subset S which is unrecognizable.**

- Here, (2) implies (1) and (4) implies (3) as undecidable set is a proper superset of unrecognizable set. (By undecidable set, I mean Set of all undecidable languages. Similarly, for unrecognizable set. NOTE that if a language is unrecognizable then it is undecidable as well, so, undecidable set is a proper superset of unrecognizable set. Decidable set is a proper subset of Recognizable set because every decidable language is recognizable.)

For regular languages, we can prove statement 1 using pumping lemma as well.

Using the pumping lemma, we find x, y, z such that xy^nz is in language L for every n, and then consider the subset $S = \{xy^nz \mid n \in A\}$, where A is any undecidable set of natural numbers. So, S is Not decidable.

S_2 : Every finite language is regular.

It is true.

Proof:

Assume that we have a finite language $L = \{w_1, w_2, w_3, \ldots, w_n\}$

For L we can write down:

Regular grammar:

- $S \rightarrow w_1 \mid w_2 \mid w_3 \mid \ldots \mid w_n$

Regular expression:

- $w_1 + w_2 + w_3 + \ldots + w_n$

Hence, L is regular.

Edit :

Some more variations :

The statement S_1 can be written in Contrapositive manner as following :

S_1 : Every infinite regular language contains an undecidable language as a subset.

Contrapositive of S_1 : If ALL the subsets of a regular language L are decidable then L is finite.

Similarly, we can say, in contrapositive manner, that :

1. If ALL the subsets of a regular language L are recognizable then L is finite.
2. If ALL the subsets of a language L are decidable then L is finite.
3. If ALL the subsets of a language L are recognizable then L is finite.

👍 6 votes -- Deepak Poonia (23.3k points)

5.5 Finite Automata (41) top

5.5.1 Finite Automata: GATE CSE 1988 | Question: 15 top ☛ https://gateoverflow.in/94642

Consider the DFA M and NFA M_2 as defined below. Let the language accepted by machine M be L. What language machine M_2 accepts, if

i. $F2 = A$?
ii. $F2 = B$?
iii. $F2 = C$?
iv. $F2 = D$?

- $M = (Q, \Sigma, \delta, q_0, F)$
- $M_2 = (Q2, \Sigma, \delta_2, q_{00}, F2)$

Where,

$Q2 = (Q \times Q \times Q) \cup \{q_{00}\}$

$\delta_2(q_{00}, \epsilon) = \{\langle q_0, q, q \rangle \mid q \in Q\}$

$\delta_2(\langle p, q, r \rangle, \sigma) = \langle \delta(p, \sigma), \delta(q, \sigma), r \rangle$

for all $p, q, r \in Q$ and $\sigma \in \Sigma$

$A = \{\langle p, q, r \rangle \mid p \in F; q, r \in Q\}$

$B = \{\langle p, q, r \rangle \mid q \in F; p, r \in Q\}$

$C = \{\langle p, q, r \rangle \mid p, q, r \in Q; \exists s \in \Sigma^* (\delta(p, s) \in F)\}$

$D = \{\langle p, q, r \rangle \mid p \in Q; q \in F\}$

normal gate1988 descriptive theory-of-computation finite-automata

Answer

5.5.2 Finite Automata: GATE CSE 1991 | Question: 17,b top ☛ https://gateoverflow.in/544

Let L be the language of all binary strings in which the third symbol from the right is a 1. Give a non-deterministic finite automaton that recognizes L. How many states does the minimized equivalent deterministic finite automaton have? Justify your answer briefly?

gate1991 theory-of-computation finite-automata normal descriptive

Answer

5.5.3 Finite Automata: GATE CSE 1993 | Question: 27 top ☛ https://gateoverflow.in/2323

Draw the state transition of a deterministic finite state automaton which accepts all strings from the alphabet $\{a, b\}$, such that no string has 3 consecutive occurrences of the letter b.

gate1993 theory-of-computation finite-automata easy descriptive

Answer

5.5.4 Finite Automata: GATE CSE 1994 | Question: 3.3 top

https://gateoverflow.in/2480

State True or False with one line explanation

A FSM (Finite State Machine) can be designed to add two integers of any arbitrary length (arbitrary number of digits).

gate1994 theory-of-computation finite-automata normal true-false

Answer

5.5.5 Finite Automata: GATE CSE 1995 | Question: 2.23 top

https://gateoverflow.in/2636

A finite state machine with the following state table has a single input x and a single out z.

present state	next state, z x=1	next state, z x=0
A	D,0	B,0
B	B,1	C,1
C	B,0	D,1
D	B,1	C,0

If the initial state is unknown, then the shortest input sequence to reach the final state C is:

A. 01
B. 10
C. 101
D. 110

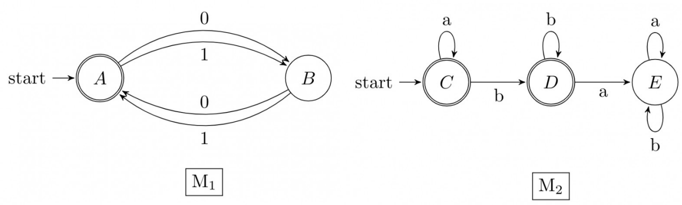
gate1995 theory-of-computation finite-automata normal

Answer

5.5.6 Finite Automata: GATE CSE 1996 | Question: 12 top

https://gateoverflow.in/2764

Given below are the transition diagrams for two finite state machines M_1 and M_2 recognizing languages L_1 and L_2 respectively.



A. Display the transition diagram for a machine that recognizes $L_1.L_2$, obtained from transition diagrams for M_1 and M_2 by adding only ε transitions and no new states.
B. Modify the transition diagram obtained in part (a) obtain a transition diagram for a machine that recognizes $(L_1.L_2)^*$ by adding only ε transitions and no new states.
   (Final states are enclosed in double circles).

gate1996 theory-of-computation finite-automata normal descriptive

Answer

5.5.7 Finite Automata: GATE CSE 1997 | Question: 21 top

https://gateoverflow.in/2281

Given that L is a language accepted by a finite state machine, show that L^P and L^R are also accepted by some finite state machines, where

$$L^P = \{s \mid ss' \in L \text{ some string } s'\}$$

$$L^R = \{s \mid s \text{ obtained by reversing some string in } L\}$$

gate1997 theory-of-computation finite-automata proof

Answer

5.5.8 Finite Automata: GATE CSE 1998 | Question: 1.10 top https://gateoverflow.in/1647

Which of the following set can be recognized by a Deterministic Finite state Automaton?

A. The numbers $1, 2, 4, 8, \dots 2^n, \dots$ written in binary
B. The numbers $1, 2, 4, 8, \dots 2^n, \dots$ written in unary
C. The set of binary string in which the number of zeros is the same as the number of ones.
D. The set $\{1, 101, 11011, 1110111, \dots\}$

gate1998 theory-of-computation finite-automata normal

Answer

5.5.9 Finite Automata: GATE CSE 2001 | Question: 5 top https://gateoverflow.in/746

Construct DFA's for the following languages:

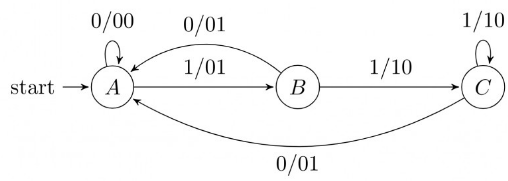
A. $L = \{w \mid w \in \{a, b\}^*,$ w has baab as a substring $\}$
B. $L = \{w \mid w \in \{a, b\}^*,$ w has an odd number of a's and an odd number of b's $\}$

gate2001-cse theory-of-computation easy descriptive finite-automata normal

Answer

5.5.10 Finite Automata: GATE CSE 2002 | Question: 2.5 top https://gateoverflow.in/835

The finite state machine described by the following state diagram with A as starting state, where an arc label is and x stands for 1-bit input and y stands for 2-bit output



A. outputs the sum of the present and the previous bits of the input
B. outputs 01 whenever the input sequence contains 11
C. outputs 00 whenever the input sequence contains 10
D. none of the above

gate2002-cse theory-of-computation normal finite-automata

Answer

5.5.11 Finite Automata: GATE CSE 2002 | Question: 21 top https://gateoverflow.in/874

We require a four state automaton to recognize the regular expression $(a \mid b)^* abb$

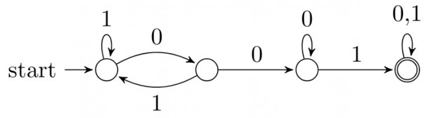
A. Give an NFA for this purpose
B. Give a DFA for this purpose

gate2002-cse theory-of-computation finite-automata normal descriptive

Answer

5.5.12 Finite Automata: GATE CSE 2003 | Question: 50 top https://gateoverflow.in/939

Consider the following deterministic finite state automaton M.



Let S denote the set of seven bit binary strings in which the first, the fourth, and the last bits are 1. The number of strings in S that are accepted by M is

A. 1
B. 5
C. 7
D. 8

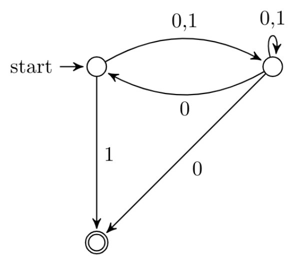
gate2003-cse theory-of-computation finite-automata normal

Answer

5.5.13 Finite Automata: GATE CSE 2003 | Question: 55 top

https://gateoverflow.in/943

Consider the NFA M shown below.



Let the language accepted by M be L. Let L_1 be the language accepted by the NFA M_1 obtained by changing the accepting state of M to a non-accepting state and by changing the non-accepting states of M to accepting states. Which of the following statements is true?

A. $L_1 = \{0,1\}^* - L$
B. $L_1 = \{0,1\}^*$
C. $L_1 \subseteq L$
D. $L_1 = L$

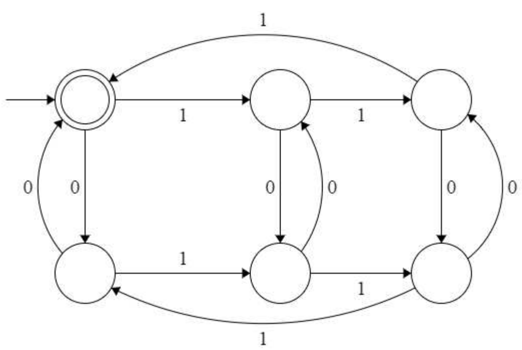
gate2003-cse theory-of-computation finite-automata normal

Answer

5.5.14 Finite Automata: GATE CSE 2004 | Question: 86 top

https://gateoverflow.in/1080

The following finite state machine accepts all those binary strings in which the number of 1's and 0's are respectively:



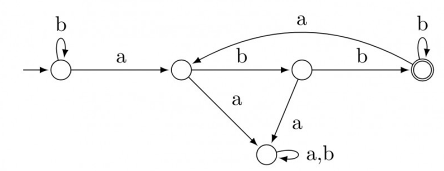
A. divisible by 3 and 2
B. odd and even
C. even and odd
D. divisible by 2 and 3

gate2004-cse theory-of-computation finite-automata easy

Answer

5.5.15 Finite Automata: GATE CSE 2005 | Question: 53 top https://gateoverflow.in/1376

Consider the machine M:



The language recognized by M is:

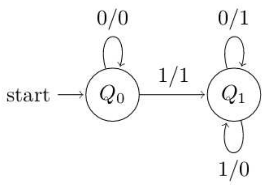
A. $\{w \in \{a, b\}^* \mid$ every a in w is followed by exactly two b's$\}$
B. $\{w \in \{a, b\}^* \mid$ every a in w is followed by at least two b's$\}$
C. $\{w \in \{a, b\}^* \mid w$ contains the substring 'abb'$\}$
D. $\{w \in \{a, b\}^* \mid w$ does not contain 'aa' as a substring$\}$

gate2005-cse theory-of-computation finite-automata normal

Answer

5.5.16 Finite Automata: GATE CSE 2005 | Question: 63 top https://gateoverflow.in/1386

The following diagram represents a finite state machine which takes as input a binary number from the least significant bit.



Which of the following is TRUE?

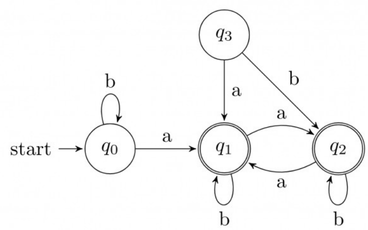
A. It computes 1's complement of the input number
B. It computes 2's complement of the input number
C. It increments the input number
D. it decrements the input number

gate2005-cse theory-of-computation finite-automata easy

Answer

5.5.17 Finite Automata: GATE CSE 2007 | Question: 74 top https://gateoverflow.in/1270

Consider the following Finite State Automaton:



The language accepted by this automaton is given by the regular expression

A. $b^*ab^*ab^*ab^*$
B. $(a+b)^*$
C. $b^*a(a+b)^*$
D. $b^*ab^*ab^*$

gate2007-cse theory-of-computation finite-automata normal

Answer

5.5.18 Finite Automata: GATE CSE 2008 | Question: 49 top

☛ https://gateoverflow.in/462

Given below are two finite state automata ($\rightarrow$ indicates the start state and F indicates a final state)

Y

	a	**b**
$\rightarrow 1$	1	2
$2(F)$	2	1

Z

	a	**b**
$\rightarrow 1$	2	2
$2(F)$	1	1

Which of the following represents the product automaton $Z \times Y$?

A.

	a	b
$\rightarrow P$	S	R
Q	R	S
R(F)	Q	P
S	Q	P

B.

	a	b
$\rightarrow P$	S	Q
Q	R	S
R(F)	Q	P
S	P	Q

C.

	a	b
$\rightarrow P$	Q	S
Q	R	S
R(F)	Q	P
S	Q	P

D.

	a	b
$\rightarrow P$	S	Q
Q	S	R
R(F)	Q	P
S	Q	P

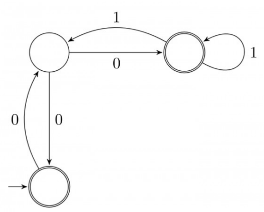
gate2008-cse normal theory-of-computation finite-automata

Answer

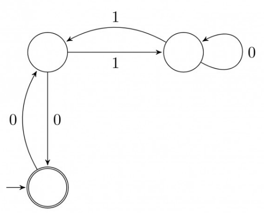
5.5.19 Finite Automata: GATE CSE 2008 | Question: 52 top

☛ https://gateoverflow.in/464

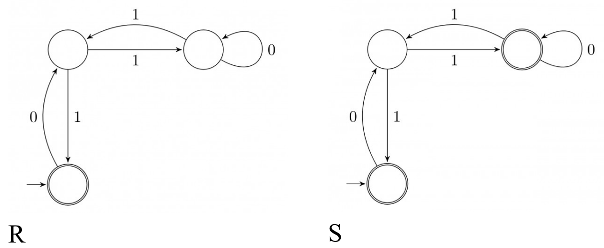
Match the following NFAs with the regular expressions they correspond to:



P



Q



1. $\epsilon + 0(01^*1 + 00)^*01^*$
2. $\epsilon + 0(10^*1 + 00)^*0$
3. $\epsilon + 0(10^*1 + 10)^*1$
4. $\epsilon + 0(10^*1 + 10)^*10^*$

A. $P-2, Q-1, R-3, S-4$
B. $P-1, Q-3, R-2, S-4$
C. $P-1, Q-2, R-3, S-4$
D. $P-3, Q-2, R-1, S-4$

gate2008-cse theory-of-computation finite-automata normal

Answer

5.5.20 Finite Automata: GATE CSE 2009 | Question: 27 top

https://gateoverflow.in/1313

Given the following state table of an FSM with two states A and B,one input and one output.

PRESENT STATE A	PRESENT STATE B	Input	Next State A	Next State B	Output
0	0	0	0	0	1
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	1	0	0
0	0	1	0	1	0
0	1	1	0	0	1
1	0	1	0	1	1
1	1	1	0	0	1

If the initial state is $A = 0, B = 0$ what is the minimum length of an input string which will take the machine to the state $A = 0, B = 1$ with $output = 1$.

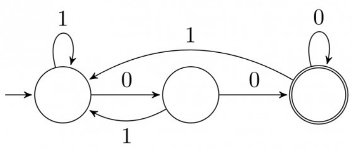
A. 3
B. 4
C. 5
D. 6

gate2009-cse theory-of-computation finite-automata normal

Answer

5.5.21 Finite Automata: GATE CSE 2009 | Question: 41 top

https://gateoverflow.in/1327



The above DFA accepts the set of all strings over $\{0, 1\}$ that

A. begin either with 0 or 1.

B. end with 0.
C. end with 00.
D. contain the substring 00.

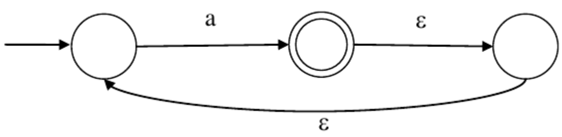
gate2009-cse theory-of-computation finite-automata easy

Answer

5.5.22 Finite Automata: GATE CSE 2012 | Question: 12 top

☛ https://gateoverflow.in/44

What is the complement of the language accepted by the NFA shown below?
Assume $\Sigma = \{a\}$ and ϵ is the empty string.



A. ϕ
B. $\{\epsilon\}$
C. a^*
D. $\{a, \epsilon\}$

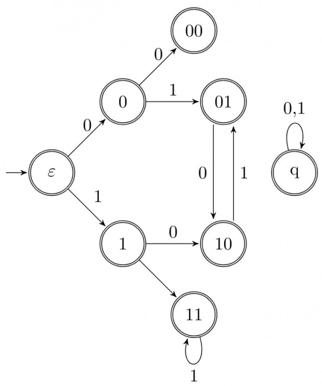
gate2012-cse finite-automata easy theory-of-computation

Answer

5.5.23 Finite Automata: GATE CSE 2012 | Question: 46 top

☛ https://gateoverflow.in/2159

Consider the set of strings on $\{0, 1\}$ in which, *every substring of* 3 *symbols* has at most *two* zeros. For example, 001110 and 011001 are in the language, but 100010 is not. All strings of length less than 3 are also in the language. A partially completed DFA that accepts this language is shown below.



The missing arcs in the DFA are:

A.

	00	01	10	11	q
00	1	0			
01				1	
10	0				
11			0		

B.

	00	01	10	11	q
00		0			1
01		1			
10				0	
11		0			

C.

	00	01	10	11	q
00		1			0
01		1			
10			0		
11		0			

D.

	00	01	10	11	q
00		1			0
01				1	
10	0				
11			0		

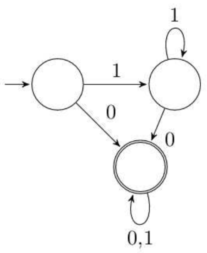
gate2012-cse theory-of-computation finite-automata normal

Answer

5.5.24 Finite Automata: GATE CSE 2013 | Question: 33 top

https://gateoverflow.in/1544

Consider the DFA A given below.



Which of the following are **FALSE**?

1. Complement of $L(A)$ is context-free.
2. $L(A) = L((11^*0 + 0)(0 + 1)^*0^*1^*)$
3. For the language accepted by A, A is the minimal DFA.
4. A accepts all strings over $\{0, 1\}$ of length at least 2.

A. 1 and 3 only
B. 2 and 4 only
C. 2 and 3 only
D. 3 and 4 only

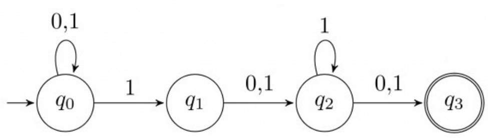
gate2013-cse theory-of-computation finite-automata normal

Answer

5.5.25 Finite Automata: GATE CSE 2014 Set 1 | Question: 16 top

https://gateoverflow.in/1782

Consider the finite automaton in the following figure:



What is the set of reachable states for the input string 0011?

A. $\{q_0, q_1, q_2\}$
B. $\{q_0, q_1\}$
C. $\{q_0, q_1, q_2, q_3\}$
D. $\{q_3\}$

gate2014-cse-set1 theory-of-computation finite-automata easy

Answer

5.5.26 Finite Automata: GATE CSE 2016 Set 2 | Question: 42 top

https://gateoverflow.in/39591

Consider the following two statements:

I. If all states of an **NFA** are accepting states then the language accepted by the **NFA** is Σ^*.
II. There exists a regular language A such that for all languages B, $A \cap B$ is regular.

Which one of the following is **CORRECT**?

A. Only **I** is true
B. Only **II** is true
C. Both **I** and **II** are true
D. Both **I** and **II** are false

gate2016-cse-set2 theory-of-computation finite-automata normal

Answer

5.5.27 Finite Automata: GATE CSE 2017 Set 2 | Question: 39 top

https://gateoverflow.in/118384

Let δ denote the transition function and $\hat{\delta}$ denote the extended transition function of the ϵ-NFA whose transition table is given below:

δ	ϵ	a	b
$\rightarrow q_0$	$\{q_2\}$	$\{q_1\}$	$\{q_0\}$
q_1	$\{q_2\}$	$\{q_2\}$	$\{q_3\}$
q_2	$\{q_0\}$	$\emptyset$	$\emptyset$
q_3	$\emptyset$	$\emptyset$	$\{q_2\}$

Then $\hat{\delta}(q_2, aba)$ is

A. $\emptyset$
B. $\{q_0, q_1, q_3\}$
C. $\{q_0, q_1, q_2\}$
D. $\{q_0, q_2, q_3\}$

gate2017-cse-set2 theory-of-computation finite-automata

Answer

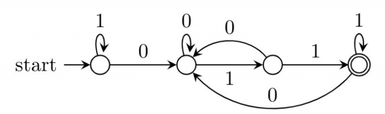
5.5.28 Finite Automata: GATE CSE 2021 Set 1 | Question: 38 top

https://gateoverflow.in/357413

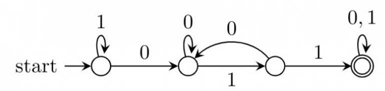
Consider the following language:

$$L = \{w \in \{0,1\}^* \mid w \text{ ends with the substring } 011\}$$

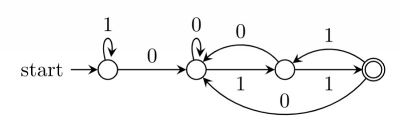
Which one of the following deterministic finite automata accepts L?



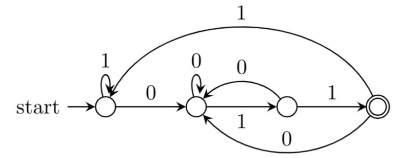
A.



B.



C.



D.

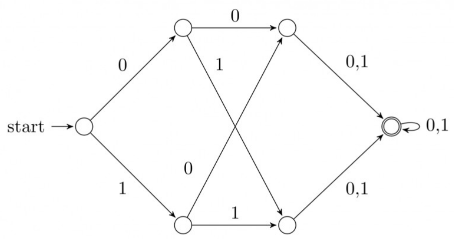
gate2021-cse-set1 theory-of-computation finite-automata

Answer

5.5.29 Finite Automata: GATE CSE 2021 Set 2 | Question: 17 top

https://gateoverflow.in/357523

Consider the following deterministic finite automaton (DFA)



The number of strings of length 8 accepted by the above automaton is ___________

gate2021-cse-set2 numerical-answers theory-of-computation finite-automata

Answer

5.5.30 Finite Automata: GATE CSE 2021 Set 2 | Question: 28 top

https://gateoverflow.in/357512

Suppose we want to design a synchronous circuit that processes a string of 0's and 1's. Given a string, it produces another string by replacing the first 1 in any subsequence of consecutive 1's by a 0. Consider the following example.

$$\begin{array}{rl} \text{Input sequence:} & 00100011000011100 \\ \text{Output sequence:} & 00000001000001100 \end{array}$$

A *Mealy Machine* is a state machine where both the next state and the output are functions of the present state and the current input.

The above mentioned circuit can be designed as a two-state Mealy machine. The states in the Mealy machine can be represented using Boolean values 0 and 1. We denote the current state, the next state, the next incoming bit, and the output bit of the Mealy machine by the variables s, t, b and y respectively.

Assume the initial state of the Mealy machine is 0.

What are the Boolean expressions corresponding to t and y in terms of s and b?

A. $t = s + b$, $y = sb$
B. $t = b$, $y = sb$
C. $t = b$, $y = s\bar{b}$
D. $t = s + b$, $y = s\bar{b}$

gate2021-cse-set2 theory-of-computation finite-automata

Answer

5.5.31 Finite Automata: GATE IT 2004 | Question: 41 top https://gateoverflow.in/3684

Let $M=(K, \Sigma, \sigma, s, F)$ be a finite state automaton, where

$K=\{A, B\}, \Sigma=\{a, b\}, s=A, F=\{B\}$,
$\sigma(A, a)=A, \sigma(A, b)=B, \sigma(B, a)=B$ and $\sigma(B, b)=A$

A grammar to generate the language accepted by M can be specified as $G=(V, \Sigma, R, S)$, where $V=K \cup \Sigma$, and $S=A$.

Which one of the following set of rules will make $L(G)=L(M)$?

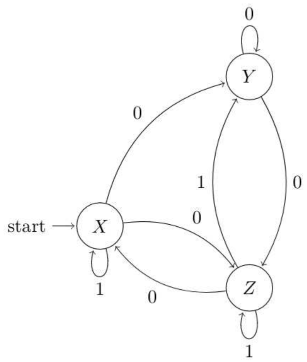
A. $\{A \rightarrow aB, A \rightarrow bA, B \rightarrow bA, B \rightarrow aA, B \rightarrow \epsilon)$
B. $\{A \rightarrow aA, A \rightarrow bB, B \rightarrow aB, B \rightarrow bA, B \rightarrow \epsilon)$
C. $\{A \rightarrow bB, A \rightarrow aB, B \rightarrow aA, B \rightarrow bA, B \rightarrow \epsilon)$
D. $\{A \rightarrow aA, A \rightarrow bA, B \rightarrow aB, B \rightarrow bA, A \rightarrow \epsilon)$

gate2004-it theory-of-computation finite-automata normal

Answer

5.5.32 Finite Automata: GATE IT 2005 | Question: 37 top https://gateoverflow.in/3784

Consider the non-deterministic finite automaton (NFA) shown in the figure.



State X is the starting state of the automaton. Let the language accepted by the NFA with Y as the only accepting state be $L1$. Similarly, let the language accepted by the NFA with Z as the only accepting state be $L2$. Which of the following statements about $L1$ and $L2$ is TRUE?

A. $L1=L2$
B. $L1 \subset L2$
C. $L2 \subset L1$
D. None of the above

gate2005-it theory-of-computation finite-automata normal

Answer

5.5.33 Finite Automata: GATE IT 2005 | Question: 39 top https://gateoverflow.in/3786

Consider the regular grammar:

- $S \rightarrow Xa \mid Ya$
- $X \rightarrow Za$
- $Z \rightarrow Sa \mid \epsilon$
- $Y \rightarrow Wa$
- $W \rightarrow Sa$

where S is the starting symbol, the set of terminals is $\{a\}$ and the set of non-terminals is $\{S, W, X, Y, Z\}$.
We wish to construct a deterministic finite automaton (DFA) to recognize the same language. What is the minimum number of states required for the DFA?

A. 2
B. 3
C. 4
D. 5

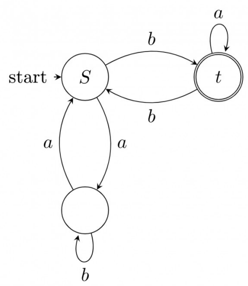
gate2005-it theory-of-computation finite-automata normal

Answer

5.5.34 Finite Automata: GATE IT 2006 | Question: 3 top

https://gateoverflow.in/3542

In the automaton below, s is the start state and t is the only final state.



Consider the strings $u = abbaba, v = bab, \text{and} w = aabb$. Which of the following statements is true?

A. The automaton accepts u and v but not w
B. The automaton accepts each of $u, v,$ and w
C. The automaton rejects each of $u, v,$ and w
D. The automaton accepts u but rejects v and w

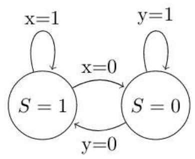
gate2006-it theory-of-computation finite-automata normal

Answer

5.5.35 Finite Automata: GATE IT 2006 | Question: 37 top

https://gateoverflow.in/3576

For a state machine with the following state diagram the expression for the next state S^+ in terms of the current state S and the input variables x and y is



A. $S^+ = S'.y' + S.x$
B. $S^+ = S.x.y' + S'.y.x'$
C. $S^+ = x.y'$
D. $S^+ = S'.y + S.x'$

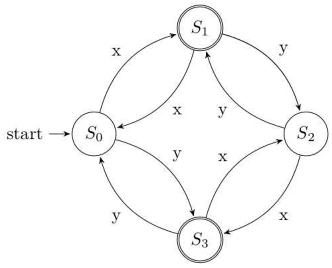
gate2006-it theory-of-computation finite-automata normal

Answer

5.5.36 Finite Automata: GATE IT 2007 | Question: 47 top

https://gateoverflow.in/3489

Consider the following DFA in which S_0 is the start state and S_1, S_3 are the final states.



What language does this DFA recognize?

A. All strings of x and y

B. All strings of x and y which have either even number of x and even number of y or odd number of x and odd number of y
C. All strings of x and y which have equal number of x and y
D. All strings of x and y with either even number of x and odd number of y or odd number of x and even number of y

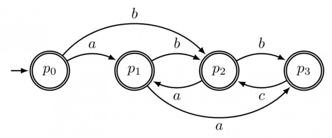
gate2007-it theory-of-computation finite-automata normal

Answer

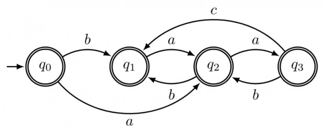
5.5.37 Finite Automata: GATE IT 2007 | Question: 50 top

☛ https://gateoverflow.in/3492

Consider the following finite automata P and Q over the alphabet $\{a, b, c\}$. The start states are indicated by a double arrow and final states are indicated by a double circle. Let the languages recognized by them be denoted by $L(P)$ and $L(Q)$ respectively.

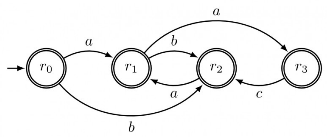


(P)

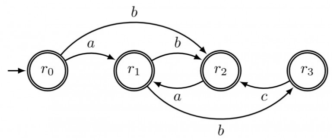


(Q)

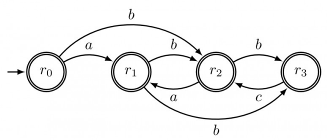
The automation which recognizes the language $L(P) \cap L(Q)$ is :



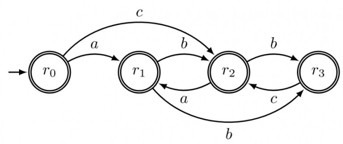
A.



B.



C.



D.

gate2007-it theory-of-computation finite-automata normal

Answer

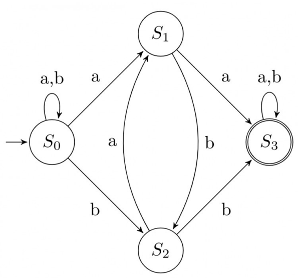
5.5.38 Finite Automata: GATE IT 2007 | Question: 71 top

☛ https://gateoverflow.in/3523

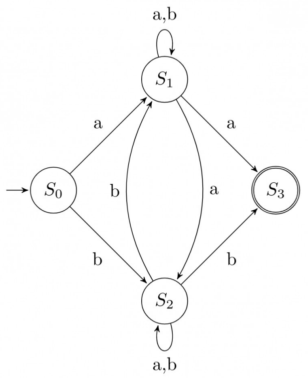
Consider the regular expression $R = (a + b)^*(aa + bb)(a + b)^*$

Which of the following non-deterministic finite automata recognizes the language defined by the regular expression R? Edges labeled λ denote transitions on the empty string.

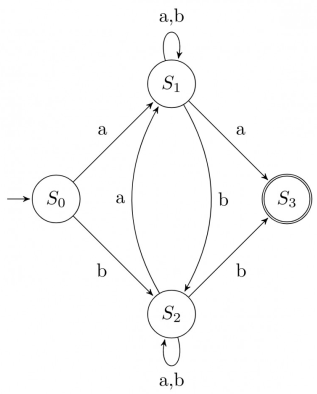
A.



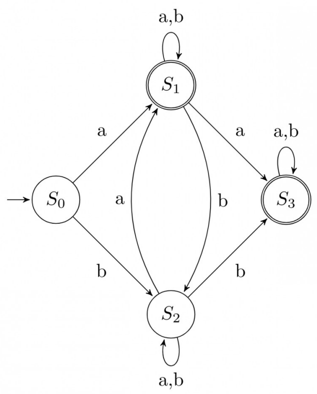
B.



C.



D.



gate2007-it theory-of-computation finite-automata normal

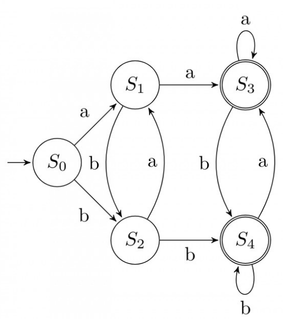
Answer

5.5.39 Finite Automata: GATE IT 2007 | Question: 72 top

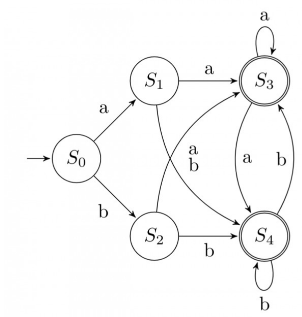
https://gateoverflow.in/3524

Consider the regular expression $R = (a+b)^*(aa+bb)(a+b)^*$

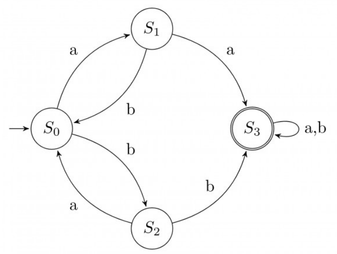
Which deterministic finite automaton accepts the language represented by the regular expression R?



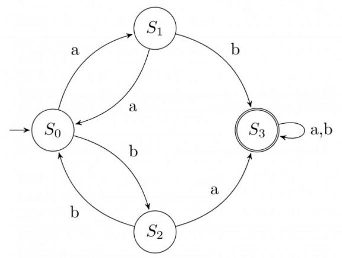
A.



B.



C.



D.

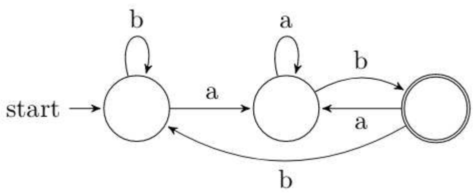
gate2007-it theory-of-computation finite-automata normal

Answer

5.5.40 Finite Automata: GATE IT 2008 | Question: 32 top

https://gateoverflow.in/3342

If the final states and non-final states in the DFA below are interchanged, then which of the following languages over the alphabet $\{a, b\}$ will be accepted by the new DFA?



A. Set of all strings that do not end with ab
B. Set of all strings that begin with either an a or a b
C. Set of all strings that do not contain the substring ab,
D. The set described by the regular expression $b^*aa^*(ba)^*b^*$

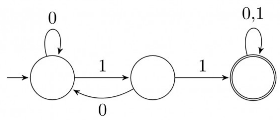
gate2008-it theory-of-computation finite-automata normal

Answer

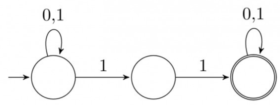
5.5.41 Finite Automata: GATE IT 2008 | Question: 36 top

https://gateoverflow.in/3346

Consider the following two finite automata. M_1 accepts L_1 and M_2 accepts L_2.



M_1



M_2

Which one of the following is TRUE?

A. $L_1 = L_2$
B. $L_1 \subset L_2$
C. $L_1 \cap L_2^C = \varnothing$
D. $L_1 \cup L_2 \neq L_1$

gatecse2008-it theory-of-computation finite-automata normal

Answer

Answers: Finite Automata

5.5.1 Finite Automata: GATE CSE 1988 | Question: 15 top

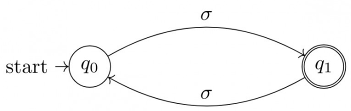
https://gateoverflow.in/94642

DFA 'M'

$M = (Q, \Sigma, \delta, q_0, F)$

Let, $Q = \{q_0, q_1\}, F = q_1$ and the transition function be defined as -

δ :



$L = \{\sigma, \sigma_3, \sigma_5 \ldots\}$ (odd number of σs)

NFA $'M_2'$

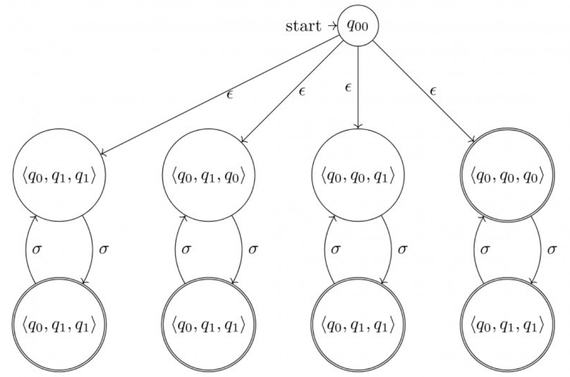
$M_2 = (Q_2, \Sigma, \delta_2, q_{00}, F_2)$

$Q_2 = \{q_{00},$

- $\langle q_0, q_0, q_0 \rangle$
- $\langle q_0, q_0, q_1 \rangle$
- $\langle q_0, q_1, q_0 \rangle$
- $\langle q_0, q_1, q_1 \rangle$
- $\langle q_1, q_0, q_0 \rangle$
- $\langle q_1, q_0, q_1 \rangle$
- $\langle q_1, q_1, q_0 \rangle$
- $\langle q_1, q_1, q_1 \rangle\}$

Now, (A) $F_2 = A = \{\langle p, q, r \rangle \mid p \in F; q, r \in Q\} \forall p, q, r \in Q \;\; \& \; \sigma \in \Sigma$

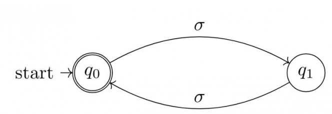
Now NFA M_2 will be

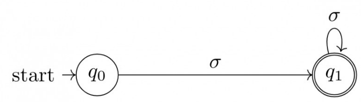


$L' = \{\sigma, \sigma_3, \sigma_5 \dots\}$

so $L' = L$ so $M_2 \equiv M$

$\Rightarrow$ we can take other transition rules for $'M'$ like,

1. 

2. 

etc or $Q = \{q_0, q_1, q_2 \dots\}$

In all cases, $L' = L$ because, we have to start with triplet $\langle q_0, , \rangle$ & then follow the path as in $'M'$ and first part of the triplet is important here because it shows the final state of NFA $'M_2'$ since $\Sigma = \{\sigma\}$ only, we have to trace first part of the triplet of M_2 which is same as tracing of DFA $'M'$.

1 votes -- ankitgupta.1729 (15k points)

5.5.2 Finite Automata: GATE CSE 1991 | Question: 17,b top

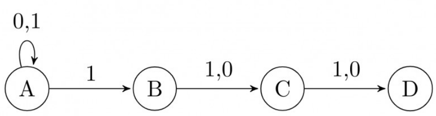
https://gateoverflow.in/544

✔ **Answer:**

Check following NFA. I've done subset construction too. 8 States are needed even after minimization.

Every state containing D is final state.

NFA:



NFA to DFA

	0	1
A	A	AB
AB	AC	ABC
AC	AD	ABD
AD	A	AB
ABC	ACD	ABCD
ABD	AC	ABC
ACD	AD	ABD
ABCD	ACD	ABCD

The third symbol from the right is a '1'. So, we can also consider the Myhill-Nerode theorem here. Intuitively we need to remember the last 3 bits of the string each of which forms a different equivalence class as per the Myhill-Nerode theorem as shown by the following table. Here, for any set of strings (in a row), we distinguish only the rows above it - as the relation is symmetric. Further strings in the language and not in the language are distinguished separately as ϵ distinguishes them.

	Last 3 bits	**Distinguishing string**	**In L?**
1	000		N
2	001	"00" distinguishes from strings in 1.	N
3	010	"0" distinguishes from strings in 1 and 2. "00" distinguishes from strings in 4.	N
4	011	"0" distinguishes from strings in 1 and 2. "00" distinguishes from strings in 3.	N
5	100		Y
6	101	"00" distinguishes from strings in 5.	Y
7	110	"0" distinguishes from strings in 5. "00" distinguishes from strings in 6.	Y
8	111	"00" distinguishes from strings in 5 and 7. "0" distinguishes from strings in 6.	Y

28 votes -- Akash Kanase (36k points)

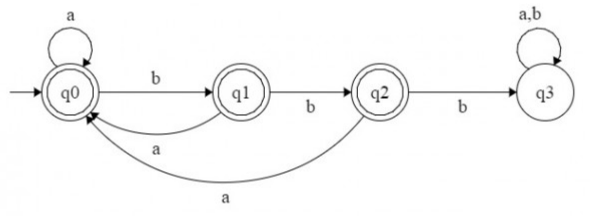
5.5.3 Finite Automata: GATE CSE 1993 | Question: 27 top

https://gateoverflow.in/2323

✔ Design a DFA that accepts all strings contain bbb

regular expression $(a+b)^* bbb (a+b)^*$

then take complement of DFA such that no string has 3 consecutive occurrences of the letter b.



having regular expression $(a+ba+bba)^*(\epsilon+b+bb)$

46 votes -- Praveen Saini (41.9k points)

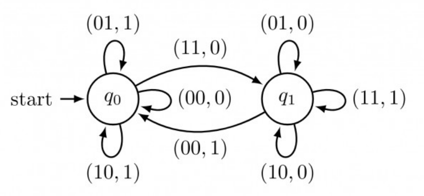
5.5.4 Finite Automata: GATE CSE 1994 | Question: 3.3 top

https://gateoverflow.in/2480

✔ Finite Automata (FA) or Finite State Machine to add two integers can be constructed using two states:

- q_0: Start state to represent carry bit is 0
- q_1: State to represent carry bit is 1

The inputs to FA will be pair of bits i.e. $00, 01, 10,$ and 11



The FA starts in-state 1 (since carry is 0) and inputs a pair of bits. If the pair is 11, the FA outputs a 0 and switches to state 2 (since the carry is 1), where the next pair of bits is input and is added to a carry bit of 1.

Example: Consider the addition of 52 and 21

- 110100 - (binary representation of 52)
- 010101 - (binary representation of 21)

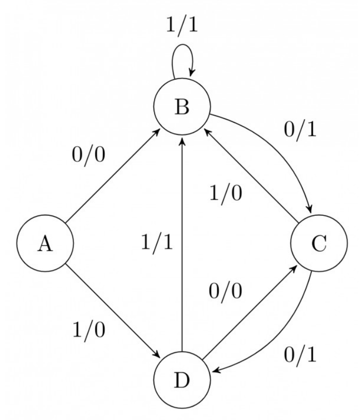
Since adding numbers is done from right to left, the first input symbol is 01, representing a 0 in the rightmost (binary) digit of 52 and a 1 in the rightmost digit of 21. The machine enters state q_0 (since there is no carry) and outputs a 1. The next input is 00 because both numbers have zero as the second rightmost digit. The machine enters state q_0 and outputs 0. The next input is 11. The machine enters state q_1 (since the carry is 1) and outputs 0. Being in-state q_1 means that there is a carry from this position into the next. And the remaining bits can be worked out to get 1001001(i.e. 73).

72 votes -- Saurav Kumar Gupta (1.7k points)

5.5.5 Finite Automata: GATE CSE 1995 | Question: 2.23 top

https://gateoverflow.in/2636

✔



- if A is start state , shortest sequence is 10 or 00 to reach C
- if B is start state , shortest sequence is 0 to reach C
- if C is start state , shortest sequence is 10 or 00 to reach C
- if D is start state , shortest sequence is 0 to reach C

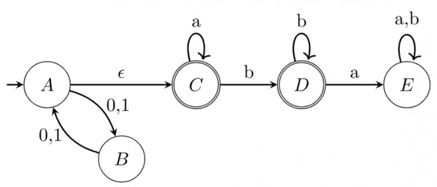
B) is correct.

49 votes -- Praveen Saini (41.9k points)

5.5.6 Finite Automata: GATE CSE 1996 | Question: 12 top

https://gateoverflow.in/2764

✔ We can combine the final state of M_1 with the start state of M_2 as follows recognizing $L_1 L_2$. But before we combine M_1 and M_2 remove the final state of M_1 as the new machine can accept L_1 also while it should accept only $L_1 . L2$.



~Pic by Praveeen Saini

30 votes -- Arjun Suresh (330k points)

5.5.7 Finite Automata: GATE CSE 1997 | Question: 21 top

https://gateoverflow.in/2281

✔ Suppose we have a finite automation for L, then we can build a finite automation for L^P by marking all the states from which final state is reachable as the final states for new automaton, the reasoning is that suppose we can reach final state f form some state q, then that means there exists some string s' that takes automation from q to f, so if there is some string s that takes automation to state q from start state this string should belong to the new language L^P. (L^P is the set of all prefix strings for the string in L)

Also, we can obtain an automation for L^R by swapping the start and final states of original automation L and by reversing all the edges in the DFA.

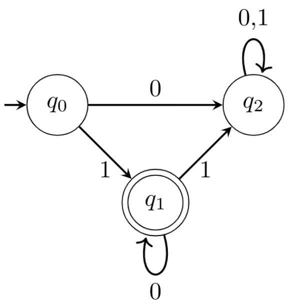
46 votes -- Omesh Pandita (1.9k points)

5.5.8 Finite Automata: GATE CSE 1998 | Question: 1.10 top https://gateoverflow.in/1647

✔ Option **A** is correct .

A. **A**. is regular
$L = \{1, 10, 100, 1000, 10000, \dots\}$
Regular expression 10^*

DFA :

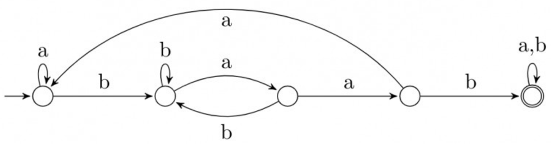


B. $L = \{1, 11, 1111, 11111111, \dots\} = \{1^i \mid i = 2^n, n \geq 0\}$ is non regular language
C. Equal - Equal is CFL, and non regular as here we have to compare the counts which need not be finite
D. $L = \{1^i01^i \mid i > 0\} \cup \{1\}$ is also CFL, and non regular

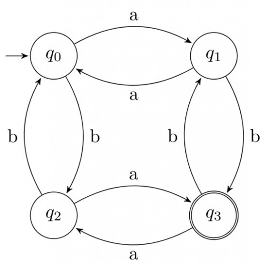
71 votes -- Praveen Saini (41.9k points)

5.5.9 Finite Automata: GATE CSE 2001 | Question: 5 top https://gateoverflow.in/746

✔ **DFA for A**:



Part (B):



30 votes -- jayendra (6.7k points)

5.5.10 Finite Automata: GATE CSE 2002 | Question: 2.5 top https://gateoverflow.in/835

✔ Answer should be option (**A**).

Option (B) and **(C)** are clearly wrong . it says input 11 then o/p 01 and i/p 10 then o/p 00 but here at single bit o/p is 2 bit sequence

Now, for option (A) we can trace out. Suppose string is 0111.

at A---0---> A----1---> B--1-->C---1-->C

O/P 00 01 10 10

We can see here at $(A, 0)$---> $(A, 00)$ which sum of $0 + 0 = 00$, (previous i/p bit + present i/p bit)

$(A, 1)$--->$(B, 01)$ which is sum of $0 + 1 = 1 = 01$,

$(B.1)$--->$(C, 10)$ which is sum of $1 + 1 =$ (previous i/p bit + present i/p bit)$= 10$,

$(C,1)$---> $(C,10)$ which is sum of $1+1=10$

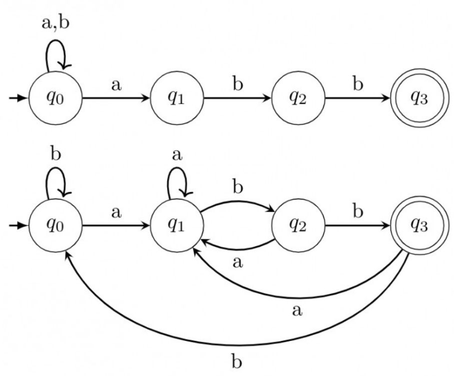
So, answer should be (**A**).

44 votes -- minal (13.1k points)

5.5.11 Finite Automata: GATE CSE 2002 | Question: 21 top

☛ https://gateoverflow.in/874

✔ **NFA** for regular expression $(a+b)^*abb$ and its equivalent DFA will be as follows:



32 votes -- Praveen Saini (41.9k points)

5.5.12 Finite Automata: GATE CSE 2003 | Question: 50 top

☛ https://gateoverflow.in/939

✔ Language of above **DFA** is all strings over $\{0,1\}$ that contain substring 001.

Regular expression of above DFA is $(0+1)^*\mathbf{001}(0+1)^*$

$\mathbf{1}$ ***that is underlined*** can not be first bit of 7-bit binary no, but can be fourth bit or last bit.

Case 1: if it is 4th bit ,then possible set of strings can be

$\mathit{First}\ 001\ \mathbf{twobits}\ \mathit{Last}\ = \mathbf{1}001(00+01+10+11)\mathbf{1} = 4$ strings

Case 2 : if it is last bit, then possible set of strings can be

$\mathit{First}\ \mathbf{twobits}\ \mathit{fourth}\ 001 = \mathbf{1}(00+01+10+11)\mathbf{1}\ 001 = 4$ strings

String common in both cases $\mathbf{1001001}$

Total strings $= 4+4-1 = 7$ strings

Correct Answer: C

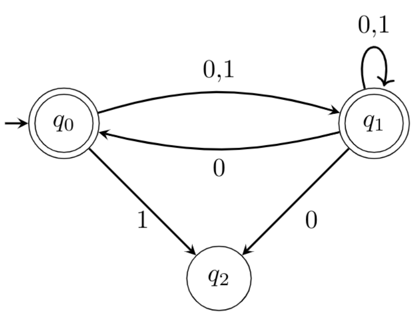
67 votes -- Praveen Saini (41.9k points)

5.5.13 Finite Automata: GATE CSE 2003 | Question: 55 top

☛ https://gateoverflow.in/943

✔ Answer is **B** .

As the problem said:



As in above **NFA** language L_1 is $\{0,1\}^*$. [we don't know L, we need not to find out]

Option A is wrong as L is accepting 1 and L_1 is also accepting 1

Option C is wrong as L_1 accepting ^,null, but L is not .

Option D is wrong for same reason as option C is wrong.

62 votes -- Praveen Saini (41.9k points)

5.5.14 Finite Automata: GATE CSE 2004 | Question: 86 top https://gateoverflow.in/1080

✔ Option **(C)** and option **(D)** are cancelled out clearly because with 3 1s we can reach the final state. There is an string where we can reach the final state by 6 1's now 6 is nt odd but it is divisivble by 3. Hence, option (**A**) is correct.

25 votes -- Bhagirathi Nayak (11.7k points)

5.5.15 Finite Automata: GATE CSE 2005 | Question: 53 top https://gateoverflow.in/1376

✔ **A** is Wrong, since $abbb$ is accepted. (1 a is followed by more than 2 b's)

C is Wrong, since $abba$ contains abb as substring, but is still not accepted.

D is Wrong, since ab does not contain aa as substring, but is still not accepted.

Hence, correct answer is **B**.

55 votes -- saurabhrk (1.1k points)

5.5.16 Finite Automata: GATE CSE 2005 | Question: 63 top https://gateoverflow.in/1386

✔ Answer is **option B.**

For any binary no, FSM read input from LSB and remain unchanged till first 1, and it complement after that

$\mathbf{100} \rightarrow \mathbf{100}$ $\quad [\text{1's complement of } 100 + 1 = 011 + 1 = 100 = \text{2's complement of } 100]$

$\mathbf{010} \rightarrow \mathbf{110}$ $\quad [\text{1's complement of } 010 + 1 = 101 + 1 = 110 = \text{2's complement of } 010]$

$1010\mathbf{100} \rightarrow 0101\mathbf{100}$ $\quad [\text{1's complement of } 1010100 + 1 = 0101011 + 1 = 0101100]$

Note : Underline part is unchanged (till first 1 from lsb) then 0's changed to 1's and 1's changed to 0's

26 votes -- Praveen Saini (41.9k points)

5.5.17 Finite Automata: GATE CSE 2007 | Question: 74 top https://gateoverflow.in/1270

✔ **The answer is C.**

You can see that both the states, q_1 and q_2 are final and are accepting $(a+b)^*$.

[Edit]

1. q_3 is unreachable state, hence it can be removed.
2. States q_1 and q_2 are indistinguishable, so, they can be merged.

39 votes -- Gate Keeda (15.9k points)

5.5.18 Finite Automata: GATE CSE 2008 | Question: 49 top https://gateoverflow.in/462

✔

	States	a	b
$\rightarrow$	**11(P)**	12	22
	12(S)	11	21
	21(Q)	22	12
(F)	**22(R)**	21	11

11 is P and 22 is R in choice. So, the answer should be (**A**) but in the row for S, it should be P and Q and not Q and P.

47 votes -- Arjun Suresh (330k points)

5.5.19 Finite Automata: GATE CSE 2008 | Question: 52 top https://gateoverflow.in/464

✔ $S-4$ is confirmed

$R-3$ is true coz everything it accepts ends with 1; this is made mandatory only by 3

this rules out option B and option D

use string 01010 and compare P Vs Q; this makes $Q - 2$ as confirmed.

Hence, **option C** is correct.

👍 32 votes -- Amar Vashishth (25.2k points)

5.5.20 Finite Automata: GATE CSE 2009 | Question: 27 top ☛ https://gateoverflow.in/1313

✔ From above table, we look at next state part

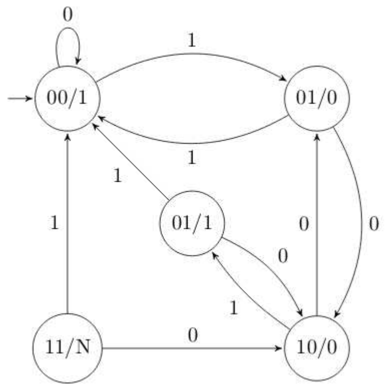
Whenever we reach state 00 we get output 1 [at row 1,row 6, row 8], ***so we have state 00 with output 1***

When we reach at state 01, we get output 0 [at row 3, row 5] and output 1 [row 7], ***so we have two state 01 with output 0, 01 with output 1.***

When we reach at state 10, we get output we get output 0 [at row 2, row 4], ***so we have state 10 with output 0.***

We don't reach at state 11 [11 is not there in next state part], but ***we have state 11 with don't know (N) output.***

If we draw the Moore Machine for above **FSM** [from the table: present state x input symbol -> next state]



It is clear from **FSM** from state 00 to reach state 01 with output 1 i.e, $01/1$ with need **minimum length input 101**

Minimum length of input $=$ length of 101. **That is 3.**

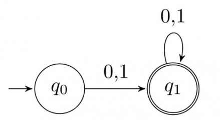
Correct Answer: A

👍 41 votes -- Praveen Saini (41.9k points)

5.5.21 Finite Automata: GATE CSE 2009 | Question: 41 top ☛ https://gateoverflow.in/1327

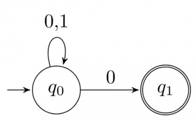
✔

A. Begin either with 0 or 1
Regular expression : $(0+1)(0+1)^*$

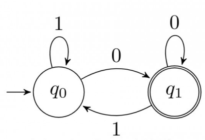


NFA $=$ DFA for begin with
0 or 1 followed by anything

B. End with 0
Regular expression : $(0+1)^*0$

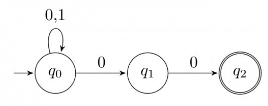


NFA for anything from
beginning, end with 0

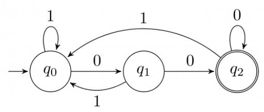


Equivalent DFA

C. End with 00
Regular expression : $(0+1)^*00$

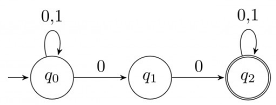


NFA for anything from beginning, end with 00

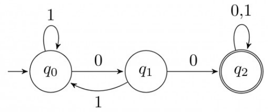


Equivalent DFA

D. Containing the substring 00
   Regular expression : $(0+1)^*00(0+1)^*$



NFA for anything followed by 00 followed by anything



Equivalent DFA

So, **C** is the correct answer.

👍 54 votes -- Praveen Saini (41.9k points)

5.5.22 Finite Automata: GATE CSE 2012 | Question: 12 top

☛ https://gateoverflow.in/44

✔ The language being accepted is a^+. So, complement of the language is $\{\epsilon\}$.

👍 69 votes -- Arjun Suresh (330k points)

5.5.23 Finite Automata: GATE CSE 2012 | Question: 46 top

☛ https://gateoverflow.in/2159

✔ (**D**) is the answer. From 00 state, a '0' should take the DFA to the dead state- q. From 11, a '0' should go to 10 representing the 10 at the end of string so far. Similarly, from 00 a 1 should go to 01, from 01 a '1' should go to 11 and from 10 a '0' should go to '00'.

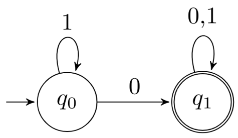
👍 36 votes -- Arjun Suresh (330k points)

5.5.24 Finite Automata: GATE CSE 2013 | Question: 33 top

☛ https://gateoverflow.in/1544

✔

1. Complement of $L(A)$ (regular language) is regular and hence also Context Free. **True**
2. Regular expression is $(11^*0+0)(0+1)^*$ **True**
3. Minimized **DFA** is:



   Both non-final states are equivalents and can be minimized
   So, 3 is **False**
4. From 3, shortest length string reached from $q0$ to $q1$ (final) is 0, so 4 is **False**

Note :

a. $(0+1)^*0^*1^* = (0+1)^* + \text{something} = (0+1)^*$
b. $(11^*0+0)(0+1)^* = (11^*+\epsilon)0(0+1)^* = 1^*0(0+1)^*$ look at Minimized DFA at 3.

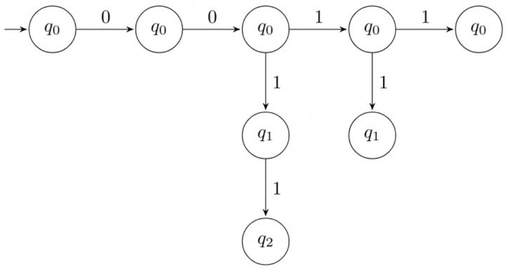
Correct Answer: D

👍 74 votes -- Praveen Saini (41.9k points)

5.5.25 Finite Automata: GATE CSE 2014 Set 1 | Question: 16 top

☛ https://gateoverflow.in/1782

✔



q_0, q_1 and q_2 are reachable from q_0 on input 0011

Correct Answer: A

👍 34 votes -- Praveen Saini (41.9k points)

5.5.26 Finite Automata: GATE CSE 2016 Set 2 | Question: 42 top

☛ https://gateoverflow.in/39591

✔

I. False, as in NFA, it is not necessary that all states have transitions for all symbols.
II. True, there exists a regular language $A = \{\}$, such that for all languages B, $A \cap B = \{\}$ is regular

So, answer is option **B**.

👍 122 votes -- Praveen Saini (41.9k points)

5.5.27 Finite Automata: GATE CSE 2017 Set 2 | Question: 39 top

☛ https://gateoverflow.in/118384

✔ Starting state : q_2 and input string is "aba"

- Step 1: Find Epsilon closure of $q_2 = \{q_2, q_0\}$
- Step 2: Find transitions on a :
  - $q_0 \to q_1$
  - $q_2 \to \emptyset$
- Step 3: Find epsilon closure of $q_1 = \{q_1, q_2, q_0\}$
- Step 4: Find transitions on b :
  - $q_1 \to q_3$
  - $q_0 \to q_0$
  - $q_2 \to \emptyset$
- Step 5: Find epsilon closure of $q_0 = \{q_0, q_2\}$ UNION epsilon closure of $q_3 = \{q_3\}, = \{q_0, q_2, q_3\}$
- Step 6: Find transitions on a :
  - $q_0 \to q_1$
  - $q_2 \to \emptyset$
  - $q_3 \to \emptyset$
- Step 7: Find epsilon closure of q_1 : $\{q_1, q_0, q_2\}$

Therefore answer is C.

👍 96 votes -- kapilthukral94 (377 points)

5.5.28 Finite Automata: GATE CSE 2021 Set 1 | Question: 38 top

☛ https://gateoverflow.in/357413

✔ The correct automata for L must accept every binary string ending with “011” and not accept any other binary string.

A. **False** it accepts binary strings ending with 111
B. **False** it accepts binary strings ending with $0, 00, 00, 100, 001, 111$ etc.
C. **False** it accepts binary string ending with 1111
D. **True** it accepts all strings that end with 011 and no other strings.

Correct Ans: **D**

👍 2 votes -- Bhargav D Dave (695 points)

5.5.29 Finite Automata: GATE CSE 2021 Set 2 | Question: 17 top

☛ https://gateoverflow.in/357523

✔ The answer is 256.

We can reach the final state with all possible strings of length three on set $\{0, 1\}$.

So there are 8 ways to reach the final state and we will have a five-length string remaining where we have two options – either 0 or 1, for each of the 5 positions.

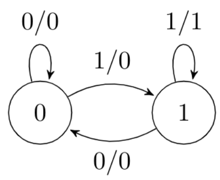
So, total number of strings accepted will be $8 \times 2^5 = 2^8 = 256$.

👍 6 votes -- JATIN MITTAL (2.1k points)

5.5.30 Finite Automata: GATE CSE 2021 Set 2 | Question: 28 top

☛ https://gateoverflow.in/357512

✔



In $(a\ /\ b)$, 'a' is input and 'b' is output

Option B is correct.

👍 3 votes -- zxy123 (2.5k points)

5.5.31 Finite Automata: GATE IT 2004 | Question: 41 top

☛ https://gateoverflow.in/3684

✔ $\sigma(A, a) = A, \qquad A \to aA$

$\sigma(A, b) = B, \qquad A \to bB$

$\sigma(B, a) = B \qquad B \to aB$

$\sigma(B, b) = A \qquad B \to bA$

B is final state so $\quad B \to \epsilon$

Correct Answer: B

👍 52 votes -- Praveen Saini (41.9k points)

5.5.32 Finite Automata: GATE IT 2005 | Question: 37 top

☛ https://gateoverflow.in/3784

✔ Misprints : Edge $Y \to Z$ (0 edge)
Edge $Z \to Y$ (1 edge)

Answer: **A.**

Explanation:

Writing Y and Z in terms of incoming arrows (Arden's method) :

$Y = X0 + Y0 + Z1$

$Z = X0 + Z1 + Y0$

Hence $Y = Z$. So, option ($\mathbf{A}$).

👍 63 votes -- Dipak Majhi (757 points)

5.5.33 Finite Automata: GATE IT 2005 | Question: 39 top

☛ https://gateoverflow.in/3786

✔

- $S \rightarrow Xa \mid Ya$
- $X \rightarrow Za$
- $Z \rightarrow Sa \mid \epsilon$
- $Y \rightarrow Wa$
- $W \rightarrow Sa$

This is left linear grammar having language $\mathbf{L}$. Convert it into right linear using following rule:

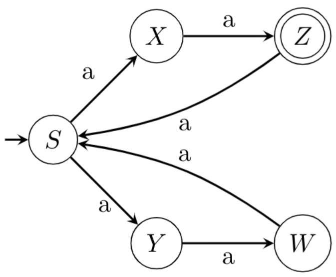
- $V_i \rightarrow V_j w \quad$ Reverses to $\quad V_i \rightarrow w^R V_j$
- $V_i \rightarrow w \quad$ Reverses to $\quad V_i \rightarrow w^R$

If tje left linear grammar produced language L then the resulting right linear grammar produces L^R.

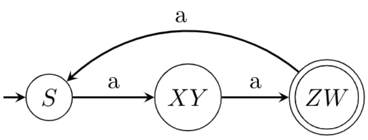
- $S \rightarrow aX \mid aY$
- $X \rightarrow aZ$
- $Z \rightarrow aS \mid \epsilon$
- $Y \rightarrow aW$
- $W \rightarrow aS$

is right linear grammar having language $\mathbf{L^R}$.

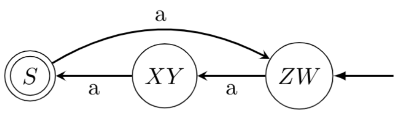
Having **NFA**



Having **DFA** for language $\mathbf{L^R}$



DFA for language L (reversal)



$\mathbf{L = \{w : n_a(w) \bmod 3 = 2,\ w \text{ belongs to } \{a, b\}^*\}}$ **same as Omesh Pandita answered.**

Having 3 states.

Correct Answer: B

72 votes -- Praveen Saini (41.9k points)

5.5.34 Finite Automata: GATE IT 2006 | Question: 3 top

☛ https://gateoverflow.in/3542

✔

for u	for v	for w
$\delta(s, abbaba)$	$\delta(s, bab)$	$\delta(s, aabb)$
$\vdash \delta(x, bbaba)$	$\vdash \delta(t, ab)$	$\vdash \delta(x, abb)$
$\vdash \delta(x, baba)$	$\vdash \delta(t, b)$	$\vdash \delta(s, bb)$
$\vdash \delta(x, aba)$	$\vdash \mathbf{s - rejected}$	$\vdash \delta(t, b)$
$\vdash \delta(s, ba)$		$\vdash \mathbf{s - rejected}$
$\vdash \delta(t, a)$		
$\vdash \mathbf{t - accepted}$		

Correct Answer: D

29 votes -- Praveen Saini (41.9k points)

5.5.35 Finite Automata: GATE IT 2006 | Question: 37 top https://gateoverflow.in/3576

Present State	Inputs		Next State
S	**x**	**y**	$\mathbf{S^+}$
0	X	0	1
0	X	1	0
1	0	X	0
1	1	X	1

From above table:

$S^+ = S'y' + Sx$

Correct Answer: A

70 votes -- Praveen Saini (41.9k points)

5.5.36 Finite Automata: GATE IT 2007 | Question: 47 top https://gateoverflow.in/3489

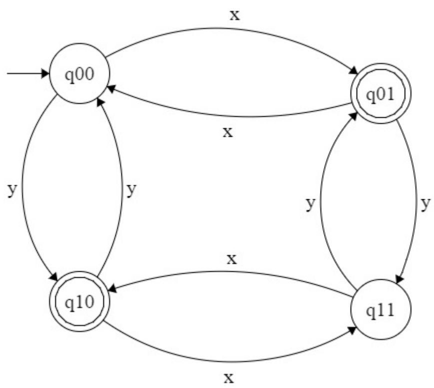
Above **DFA** can be redesigned as: [S_0 as $q00$, S_1 as $q10$, S_2 as $q11$, S_3 as $q01$]

Where, each state as $q_{ab}[a = n_a \ mod \ 2, b = n_b \ mod \ 2]$

$q00$ as $n_a \ mod \ 2 = 0, n_b \ mod \ 2 = 0$ [no of x is even no of y is even]

and $\delta(q00, x) \rightarrow q10$ $[(0+1) \ mod \ 2 = 1$ as x increase from 0 to 1] $\delta(q00, y) \rightarrow q01$

and $\delta(q10, x) \rightarrow q00[(1+1) \ mod \ 2 = 0]$ $\delta(q00, y) \rightarrow q01$ and soon



$q01$ is final state mean where no of x is even and no of y is odd

$q10$ is final state mean where no of x is odd and no of y is even.

So, **D** is correct answer.

27 votes -- Praveen Saini (41.9k points)

5.5.37 Finite Automata: GATE IT 2007 | Question: 50 top https://gateoverflow.in/3492

Design an automaton using P and Q having p_0q_0 as start state

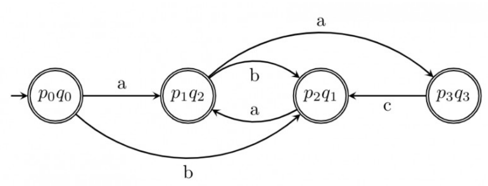
$\delta(p_0q_0, a) \rightarrow \delta(p_0, a) \cup \delta(q_0, a)$

$Q \setminus \Sigma$	a	b	c
$\rightarrow p_0q_0^*$	p_1q_2	p_2q_1	
$p_1q_2^*$	p_3q_3		
$p_2q_1^*$	p_1q_2	p_3 (No Need)	
$p_3q_3^*$		q_2 (No Need)	p_2q_1

In case of intersection final states are those where final states of P and final states of Q come together.

"No need" in above table mean, when we reach to p_3 (or q_2) then we cannot reach to any final state because we cannot have states of P and Q together (intersection). Hence, this is not shown in diagram [may draw a dead state for it to make it a DFA (as noted at end)]

Automaton results in:



That is option **A** .

Note : DFA must have transition for each symbol $Q \times \Sigma \rightarrow Q$ and hence our automaton is not a DFA as we do not have transitions to dead state.

76 votes -- Praveen Saini (41.9k points)

See the languages being accepted by P and Q. In P before a 'c' there must be either 'b' or 'aa'. In Q, before 'c' there must be 'aa'. So, obviously, in their intersection before 'c' there must be 'aa' which is satisfied only by option A.

39 votes -- Arjun Suresh (330k points)

5.5.38 Finite Automata: GATE IT 2007 | Question: 71 top

https://gateoverflow.in/3523

✔ When we say non-deterministic finite automata recognizes the language defined by the regular expression R then it means that it won't accept any other string that does not fall under the R and it will only accept all strings that fall under R. R says the string should contain either $\mathbf{aa}$ or $\mathbf{bb}$.

So, we see **B** accepts $\mathbf{aba}$, **C** accepts $\mathbf{aba}$ and **D** accepts $\mathbf{a}$. Just find some examples that do not follow the rule. **A** accepts any string containing either $\mathbf{aa}$ or $\mathbf{bb}$ and it does not accept any other string.

Option **A** is correct.

45 votes -- Shreya Roy (3.8k points)

5.5.39 Finite Automata: GATE IT 2007 | Question: 72 top

https://gateoverflow.in/3524

✔ **DFA** given in option **A**

Here, S_3 and S_4 are equivalent states and can be minimized.

This results in **DFA** given in:

https://gateoverflow.in/3523/gate2007-it_71

References

25 votes -- Praveen Saini (41.9k points)

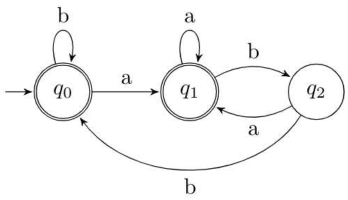
5.5.40 Finite Automata: GATE IT 2008 | Question: 32 top

https://gateoverflow.in/3342

✔ Above **DFA** is for regular expression $(a + b)^*ab$. All strings end with ab.

Complement of **DFA** accepts all strings does not end with ab.

DFA(L') is:



B. String begin with either a or b.

ab (string start with a) doesn't accept in it reach to nonfinal state q_2.

bab (string start with b) doesn't accept in it reach to nonfinal state q_2.

C. Set of strings that do not contain the substring ab

aba (have substring ab) does accept in it reach to final state $q1$.

D. The set described by the regular expression $b^*aa^*(ba)^*b^*$

b is string accepted by DFA(L') but above regular expression cannot derive it.

Option **A** is correct.

DFA (L') accepts all strings that doesn't end with ab.

42 votes -- Praveen Saini (41.9k points)

5.5.41 Finite Automata: GATE IT 2008 | Question: 36 top https://gateoverflow.in/3346

✔ $L_1 : (0+10)^*11(0+1)^*$

$L_2 : (0+1)^*11(0+1)^*$

It is quite clear that $L_1 = L_2$.

As both languages L_1 and L_2 are equal So Complement of Language L_2 will be the complement of Language L_1 also. For a given language $L, L \cap L^c = \emptyset$.

Hence, both options (**A**) and (**C**) are correct.

39 votes -- Vicky Bajoria (4.1k points)

5.6 Grammar (1) top

5.6.1 Grammar: GATE CSE 2008 | Question: 51 top https://gateoverflow.in/463

Match the following:

E.	Checking that identifiers are declared before their use	P.	$L = \{a^n b^m c^n d^m \mid n \geq 1, m \geq 1\}$
F.	Number of formal parameters in the declaration of a function agrees with the number of actual parameters in a use of that function	Q.	$X \rightarrow XbX \mid XcX \mid dXf \mid g$
G.	Arithmetic expressions with matched pairs of parentheses	R.	$L = \{wcw \mid w \in (a \mid b)^*\}$
H.	Palindromes	S.	$X \rightarrow bXb \mid cXc \mid \epsilon$

A. E-P, F-R, G-Q, H-S
B. E-R, F-P, G-S, H-Q
C. E-R, F-P, G-Q, H-S
D. E-P, F-R, G-S, H-Q

gate2008-cse normal theory-of-computation grammar

Answer

Answers: Grammar

5.6.1 Grammar: GATE CSE 2008 | Question: 51 top

https://gateoverflow.in/463

✔ H-S is true because strings generated by this grammar satisfies the definition of an even length palindrome string. This rules out B and D options.

G-Q is confirmed as both options A and C has it as true.

E-R is true because R is the only grammar that checks: what (w) has occurred earlier is present afterwards. This equals the definition of E

Hence, **option C** is true.

49 votes -- Amar Vashishth (25.2k points)

5.7 Identify Class Language (27) top

5.7.1 Identify Class Language: GATE CSE 1987 | Question: 1-xiii top

https://gateoverflow.in/80293

FORTRAN is a:

A. Regular language.
B. Context-free language.
C. Context-senstive language.
D. None of the above.

gate1987 theory-of-computation identify-class-language

Answer

5.7.2 Identify Class Language: GATE CSE 1988 | Question: 2ix top

https://gateoverflow.in/93949

What is the type of the language L, where $L = \{a^n b^n \mid 0 < n < 327\text{-th prime number}\}$

gate1988 normal descriptive theory-of-computation identify-class-language

Answer

5.7.3 Identify Class Language: GATE CSE 1991 | Question: 17,a top

https://gateoverflow.in/26653

Show that the Turing machines, which have a read only input tape and constant size work tape, recognize precisely the class of regular languages.

gate1991 theory-of-computation descriptive identify-class-language proof

Answer

5.7.4 Identify Class Language: GATE CSE 1994 | Question: 19 top

https://gateoverflow.in/2515

A. Given a set:

$$S = \{x \mid \text{ there is an x-block of 5's in the decimal expansion of } \pi\}$$

(Note: x-*block* is a maximal block of x successive 5's)
Which of the following statements is true with respect to S? No reason to be given for the answer.

i. S is regular
ii. S is recursively enumerable
iii. S is not recursively enumerable
iv. S is recursive

B. Given that a language L_1 is regular and that the language $L_1 \cup L_2$ is regular, is the language L_2 always regular? Prove your answer.

gate1994 theory-of-computation identify-class-language normal descriptive

Answer

5.7.5 Identify Class Language: GATE CSE 1999 | Question: 2.4 top

https://gateoverflow.in/1482

If $L1$ is context free language and $L2$ is a regular language which of the following is/are false?

A. $L1 - L2$ is not context free
B. $L1 \cap L2$ is context free
C. $\sim L1$ is context free
D. $\sim L2$ is regular

gate1999 theory-of-computation identify-class-language normal multiple-selects

Answer

5.7.6 Identify Class Language: GATE CSE 2000 | Question: 1.5 top

https://gateoverflow.in/628

Let L denote the languages generated by the grammar $S \rightarrow 0S0 \mid 00$.
Which of the following is TRUE?

A. $L = 0^+$
B. L is regular but not 0^+
C. L is context free but not regular
D. L is not context free

gate2000-cse theory-of-computation easy identify-class-language

Answer

5.7.7 Identify Class Language: GATE CSE 2002 | Question: 1.7 top

https://gateoverflow.in/811

The language accepted by a Pushdown Automaton in which the stack is limited to 10 items is best described as

A. Context free
B. Regular
C. Deterministic Context free
D. Recursive

gate2002-cse theory-of-computation easy identify-class-language

Answer

5.7.8 Identify Class Language: GATE CSE 2004 | Question: 87 top

https://gateoverflow.in/1081

The language $\{a^m b^n c^{m+n} \mid m, n \geq 1\}$ is

A. regular
B. context-free but not regular
C. context-sensitive but not context free
D. type-0 but not context sensitive

gate2004-cse theory-of-computation normal identify-class-language

Answer

5.7.9 Identify Class Language: GATE CSE 2005 | Question: 55 top

https://gateoverflow.in/1378

Consider the languages:
$L_1 = \{a^n b^n c^m \mid n, m > 0\}$ and $L_2 = \{a^n b^m c^m \mid n, m > 0\}$
Which one of the following statements is FALSE?

A. $L_1 \cap L_2$ is a context-free language
B. $L_1 \cup L_2$ is a context-free language
C. L_1 and L_2 are context-free languages
D. $L_1 \cap L_2$ is a context sensitive language

gate2005-cse theory-of-computation identify-class-language normal

Answer

5.7.10 Identify Class Language: GATE CSE 2006 | Question: 30 top

https://gateoverflow.in/993

For $s \in (0+1)^*$ let $d(s)$ denote the decimal value of s (e.g. $d(101) = 5$). Let

$$L = \{s \in (0+1)^* \mid d(s) \bmod 5 = 2 \text{ and } d(s) \bmod 7 \neq 4\}$$

Which one of the following statements is true?

A. L is recursively enumerable, but not recursive
B. L is recursive, but not context-free
C. L is context-free, but not regular
D. L is regular

gate2006-cse theory-of-computation normal identify-class-language

Answer

5.7.11 Identify Class Language: GATE CSE 2006 | Question: 33 top

https://gateoverflow.in/996

Let L_1 be a regular language, L_2 be a deterministic context-free language and L_3 a recursively enumerable, but not recursive, language. Which one of the following statements is false?

A. $L_1 \cap L_2$ is a deterministic CFL
B. $L_3 \cap L_1$ is recursive
C. $L_1 \cup L_2$ is context free
D. $L_1 \cap L_2 \cap L_3$ is recursively enumerable

gate2006-cse theory-of-computation normal identify-class-language

Answer

5.7.12 Identify Class Language: GATE CSE 2007 | Question: 30 top

https://gateoverflow.in/1228

The language $L = \left\{0^i 21^i \mid i \geq 0\right\}$ over the alphabet $\{0, 1, 2\}$ is:

A. not recursive
B. is recursive and is a deterministic CFL
C. is a regular language
D. is not a deterministic CFL but a CFL

gate2007-cse theory-of-computation normal identify-class-language

Answer

5.7.13 Identify Class Language: GATE CSE 2008 | Question: 9 top

https://gateoverflow.in/407

Which of the following is true for the language

$$\{a^p \mid p \text{ is a prime }\}?$$

A. It is not accepted by a Turing Machine
B. It is regular but not context-free
C. It is context-free but not regular
D. It is neither regular nor context-free, but accepted by a Turing machine

gate2008-cse theory-of-computation easy identify-class-language

Answer

5.7.14 Identify Class Language: GATE CSE 2009 | Question: 40 top https://gateoverflow.in/1326

Let $L = L_1 \cap L_2$, where L_1 and L_2 are languages as defined below:

$L_1 = \{a^m b^m c a^n b^n \mid m, n \geq 0\}$

$L_2 = \{a^i b^j c^k \mid i, j, k \geq 0\}$

Then L is

A. Not recursive
B. Regular
C. Context free but not regular
D. Recursively enumerable but not context free.

gate2009-cse theory-of-computation easy identify-class-language

Answer

5.7.15 Identify Class Language: GATE CSE 2010 | Question: 40 top https://gateoverflow.in/2341

Consider the languages

$L1 = \{0^i 1^j \mid i \neq j\}$,

$L2 = \{0^i 1^j \mid i = j\}$,

$L3 = \{0^i 1^j \mid i = 2j + 1\}$,

$L4 = \{0^i 1^j \mid i \neq 2j\}$

A. Only $L2$ is context free.
B. Only $L2$ and $L3$ are context free.
C. Only $L1$ and $L2$ are context free.
D. All are context free

gate2010-cse theory-of-computation context-free-languages identify-class-language normal

Answer

5.7.16 Identify Class Language: GATE CSE 2011 | Question: 26 top https://gateoverflow.in/2128

Consider the languages $L1$, $L2$ and $L3$ as given below.

$L1 = \{0^p 1^q \mid p, q \in N\}, L2 = \{0^p 1^q \mid p, q \in N \; and \; p = q\} \; and L3 = \{0^p 1^q 0^r \mid p, q, r \in N \; and \; p = q = r\}$.

Which of the following statements is **NOT TRUE**?

A. Push Down Automata (PDA) can be used to recognize $L1$ and $L2$
B. $L1$ is a regular language
C. All the three languages are context free
D. Turing machines can be used to recognize all the languages

gate2011-cse theory-of-computation identify-class-language normal

Answer

5.7.17 Identify Class Language: GATE CSE 2013 | Question: 32 top https://gateoverflow.in/1543

Consider the following languages.

$L_1 = \{0^p 1^q 0^r \mid p, q, r \geq 0\}$

$L_2 = \{0^p 1^q 0^r \mid p, q, r \geq 0, p \neq r\}$

Which one of the following statements is **FALSE**?

A. L_2 is context-free.
B. $L_1 \cap L_2$ is context-free.
C. Complement of L_2 is recursive.
D. Complement of L_1 is context-free but not regular.

gate2013-cse theory-of-computation identify-class-language normal

Answer

5.7.18 Identify Class Language: GATE CSE 2014 Set 3 | Question: 36 top https://gateoverflow.in/2070

Consider the following languages over the alphabet $\sum = \{0, 1, c\}$

$L_1 = \{0^n 1^n \mid n \geq 0\}$

$L_2 = \{wcw^r \mid w \in \{0, 1\}^*\}$

$L_3 = \{ww^r \mid w \in \{0, 1\}^*\}$

Here, w^r is the reverse of the string w. Which of these languages are deterministic Context-free languages?

A. None of the languages
B. Only L_1
C. Only L_1 and L_2
D. All the three languages

gate2014-cse-set3 theory-of-computation identify-class-language context-free-languages normal

Answer

5.7.19 Identify Class Language: GATE CSE 2017 Set 1 | Question: 37 top https://gateoverflow.in/118320

Consider the context-free grammars over the alphabet $\{a, b, c\}$ given below. S and T are non-terminals.

$G_1 : S \rightarrow aSb \mid T, T \rightarrow cT \mid \epsilon$

$G_2 : S \rightarrow bSa \mid T, T \rightarrow cT \mid \epsilon$

The language $L\left(G_1\right) \cap L(G_2)$ is

A. Finite
B. Not finite but regular
C. Context-Free but not regular
D. Recursive but not context-free

gate2017-cse-set1 theory-of-computation context-free-languages identify-class-language normal

Answer

5.7.20 Identify Class Language: GATE CSE 2017 Set 2 | Question: 40 top https://gateoverflow.in/118615

Consider the following languages.

- $L_1 = \{a^p \mid p \text{ is a prime number}\}$
- $L_2 = \{a^n b^m c^{2m} \mid n \geq 0, m \geq 0\}$
- $L_3 = \{a^n b^n c^{2n} \mid n \geq 0\}$
- $L_4 = \{a^n b^n \mid n \geq 1\}$

Which of the following are CORRECT?

I. L_1 is context free but not regular
II. L_2 is not context free
III. L_3 is not context free but recursive
IV. L_4 is deterministic context free

A. I, II and IV only
B. II and III only
C. I and IV only
D. III and IV only

gate2017-cse-set2 theory-of-computation identify-class-language

Answer

5.7.21 Identify Class Language: GATE CSE 2018 | Question: 35 top https://gateoverflow.in/204109

Consider the following languages:

I. $\{a^m b^n c^p d^q \mid m+p=n+q, \text{ where } m,n,p,q \geq 0\}$
II. $\{a^m b^n c^p d^q \mid m=n \text{ and } p=q, \text{ where } m,n,p,q \geq 0\}$
III. $\{a^m b^n c^p d^q \mid m=n=p \text{ and } p \neq q, \text{ where } m,n,p,q \geq 0\}$
IV. $\{a^m b^n c^p d^q \mid mn=p+q, \text{ where } m,n,p,q \geq 0\}$

Which of the above languages are context-free?

A. I and IV only
B. I and II only
C. II and III only
D. II and IV only

gate2018-cse theory-of-computation identify-class-language context-free-languages normal

Answer

5.7.22 Identify Class Language: GATE CSE 2020 | Question: 10 top

https://gateoverflow.in/333221

Consider the language $L=\{a^n \mid n \geq 0\} \cup \{a^n b^n \mid n \geq 0\}$ and the following statements.

I. L is deterministic context-free.
II. L is context-free but not deterministic context-free.
III. L is not $LL(k)$ for any k.

Which of the above statements is/are TRUE?

A. I only
B. II only
C. I and III only
D. III only

gate2020-cse theory-of-computation identify-class-language

Answer

5.7.23 Identify Class Language: GATE CSE 2020 | Question: 32 top

https://gateoverflow.in/333199

Consider the following languages.

$$L_1=\{wxyx \mid w,x,y \in (0+1)^+\}$$
$$L_2=\{xy \mid x,y \in (a+b)^*, |x|=|y|, x \neq y\}$$

Which one of the following is TRUE?

A. L_1 is regular and L_2 is context- free.
B. L_1 context- free but not regular and L_2 is context-free.
C. Neither L_1 nor L_2 is context- free.
D. L_1 context- free but L_2 is not context-free.

gate2020-cse theory-of-computation identify-class-language

Answer

5.7.24 Identify Class Language: GATE CSE 2021 Set 2 | Question: 12 top

https://gateoverflow.in/357528

Let L_1 be a regular language and L_2 be a context-free language. Which of the following languages is/are context-free?

A. $L_1 \cap \overline{L_2}$
B. $\overline{\overline{L_1} \cup \overline{L_2}}$
C. $L_1 \cup (L_2 \cup \overline{L_2})$
D. $(L_1 \cap L_2) \cup (\overline{L_1} \cap L_2)$

gate2021-cse-set2 multiple-selects theory-of-computation identify-class-language

Answer

5.7.25 Identify Class Language: GATE IT 2005 | Question: 4 top https://gateoverflow.in/3748

Let L be a regular language and M be a context-free language, both over the alphabet Σ. Let L^c and M^c denote the complements of L and M respectively. Which of the following statements about the language $L^c \cup M^c$ is TRUE?

A. It is necessarily regular but not necessarily context-free.
B. It is necessarily context-free.
C. It is necessarily non-regular.
D. None of the above

gate2005-it theory-of-computation normal identify-class-language

Answer

5.7.26 Identify Class Language: GATE IT 2005 | Question: 6 top https://gateoverflow.in/3751

The language $\{0^n 1^n 2^n \mid 1 \leq n \leq 10^6\}$ is

A. regular
B. context-free but not regular
C. context-free but its complement is not context-free
D. not context-free

gate2005-it theory-of-computation easy identify-class-language

Answer

5.7.27 Identify Class Language: GATE IT 2008 | Question: 33 top https://gateoverflow.in/3343

Consider the following languages.

- $L_1 = \{a^i b^j c^k \mid i = j, k \geq 1\}$
- $L_2 = \{a^i b^j \mid j = 2i, i \geq 0\}$

Which of the following is true?

A. L_1 is not a CFL but L_2 is
B. $L_1 \cap L_2 = \varnothing$ and L_1 is non-regular
C. $L_1 \cup L_2$ is not a CFL but L_2 is
D. There is a 4-state PDA that accepts L_1, but there is no DPDA that accepts L_2.

gate2008-it theory-of-computation normal identify-class-language

Answer

Answers: Identify Class Language

5.7.1 Identify Class Language: GATE CSE 1987 | Question: 1-xiii top https://gateoverflow.in/80293

✔ **C**. Context-sensitive language.

Due to the presence of some features which cannot be handled by **PDA.**

Some of the features are:

1. Variable declared before use
2. Matching formal and actual parameters of functions

16 votes -- Ravit Anand (165 points)

5.7.2 Identify Class Language: GATE CSE 1988 | Question: 2ix top https://gateoverflow.in/93949

✔ Here n is finite and finite language must be regular.

If n is not restricted, then it would be **DCFL.**

20 votes -- kirti singh (2.6k points)

5.7.3 Identify Class Language: GATE CSE 1991 | Question: 17,a top https://gateoverflow.in/26653

✔ A read-only Turing machine or Two-way deterministic finite-state automaton (2DFA)is class of models of computability that behave like a standard Turing machine and can move in both directions across input, except cannot write to its input tape. The machine in its bare form is equivalent to a Deterministic finite automaton in computational power, and therefore can only parse a regular language.

https://en.wikipedia.org/wiki/Read-only_Turing_machine

References

9 votes -- srestha (85.2k points)

5.7.4 Identify Class Language: GATE CSE 1994 | Question: 19 top https://gateoverflow.in/2515

✔

B. No. Need not be. Take $L_2 = \{a^n b^n \mid n > 0\}$ and $L_1 =$ all strings over $\{a, b\}$. Now, $L_1 \cup L_2$ is L_1 only and is regular but L_2 is not regular.

22 votes -- Arjun Suresh (330k points)

5.7.5 Identify Class Language: GATE CSE 1999 | Question: 2.4 top https://gateoverflow.in/1482

✔ L_2 is regular, so complement of $L2, (\sim L2)$, is also regular .

Regular languages under complement. So**, D is True.**

$L_1 \cap L_2$ is context free.

Intersection of Context free language with Regular language is Context free. So, **B is True.**

$L_1 - L_2 = L_1 \cap (\sim L_2)$ is context free

Intersection of Context free language with Regular language is Context free. So, **A is False .**

$\sim L_1$ is not context free

Context free languages are not closed under complement. So **C is False (May/not be).**

36 votes -- Praveen Saini (41.9k points)

5.7.6 Identify Class Language: GATE CSE 2000 | Question: 1.5 top https://gateoverflow.in/628

✔ Correct Option: **B**

Language generated by this grammar is $L = (00)^+$ that is regular but not 0^+

40 votes -- Manu Thakur (34.1k points)

5.7.7 Identify Class Language: GATE CSE 2002 | Question: 1.7 top https://gateoverflow.in/811

✔ Correct Option: **B**

With only finite positions in stack, we can have only finite configurations and these can also be modeled as states in a finite automata.

36 votes -- Arjun Suresh (330k points)

5.7.8 Identify Class Language: GATE CSE 2004 | Question: 87 top https://gateoverflow.in/1081

✔ Language is not regular bcoz we need to match count of c's is equal to count of b's $+$ count of a's

and that can implement by **PDA.**

$\delta(q_0, a, \epsilon) = (q_0, a)$ [push a in stack, as per language a comes first]

$\delta(q_0, a, a) = (q_0, aa)$ [push all a's into stack]

$\delta(q_0, b, a) = (q_1, ba)$ [push b in stack, state change to q_1 that sure b comes after a]

$\delta(q_1, b, b) = (q_1, bb)$ [push all b's in stack]

$\delta(q_1, c, b) = (q_2, \epsilon)$ [pop one b for one c]

$\delta(q_2, c, b) = (q_2, \epsilon)$ [pop one b's for each c and continue same]

$\delta(q_2, c, a) = (q_3, \epsilon)$ [pop one a for one c , when there is no more b in stack]

$\delta(q_3, c, a) = (q_3, \epsilon)$ [pop one a for each c and continue same]

$\delta(q_3, \epsilon, \epsilon) = (q_f, \epsilon)$ [if sum of c's is sum of a's and b's then stack will be empty when there is no c in input]

Answer is **option B** : language is context- free but not regular.

Note :1state changes make sure b's comes after a and c's comes after b's]

24 votes -- Praveen Saini (41.9k points)

5.7.9 Identify Class Language: GATE CSE 2005 | Question: 55 top https://gateoverflow.in/1378

✔ L_1 is **CFL** [put a's in stack , and pop a with each b]]
L_2 is **CFL** [put b's in stack and pop b with each c]
C) is True.
B) is True. **CFL** is closed under Union [$S \to S_1 \mid S_2$ where S_1 is grammar for L_1 and S_2 for L_2]
CFL is not closed under Intersection, so intersection of two CFLs may or may not be CFL. Lets examine:
$L_1 \cap L_2 = \{a^i b^i c^i, i > 0\}$ which is Context sensitive but not context free [can't match a's,b's and c's with one stack]
So**, A)** is False**.**
D) is True.

31 votes -- Praveen Saini (41.9k points)

5.7.10 Identify Class Language: GATE CSE 2006 | Question: 30 top https://gateoverflow.in/993

✔ $L_1 = \{s \in (0+1)^* \mid d(s) \mod 5 = 2\}$ is regular

Having 2 as final state out of $\{0, 1, 2, 3, 4\}$ states

As given in example in posted link [in same DFA , final state will be 2 instead of 0]

Similarly, $L_2 = \{s \in (0+1)^* \mid d(s) \mod 7 \neq 4\}$ is also regular

Having states $\{0, 1, 2, 3, 4, 5, 6\}$ and all are final state except 4

$L_1 \cap L_2$ is L (given problem) is also regular

As regular languages are closed under intersection. **D** is correct option.
Reference: https://gateoverflow.in/1695/gate1998_4
References

60 votes -- Praveen Saini (41.9k points)

5.7.11 Identify Class Language: GATE CSE 2006 | Question: 33 top https://gateoverflow.in/996

✔
A. True : DCFL are closed under Intersection with Regular Languages
B. False : L_1 is recursive hence also decidable, L_3 is RE but not Recursive hence it is undecidable. Intersection of Recursive language and Recursive Enumerable language is Recursive Enumerable language.
C. True : L_1 is regular hence also CFL and every DCFL is also CFL and All CFL are closed under Union.
D. True : L_1 is regular hence also RE; L_2 is DCFL hence also RE; RE languages are closed under Intersection

👍 53 votes -- Danish (3.4k points)

5.7.12 Identify Class Language: GATE CSE 2007 | Question: 30 top ☛ https://gateoverflow.in/1228

✔ $L = \left\{0^i 21^i \mid i \geq 0\right\}$ has only one comparison that can be done using a DPDA. Hence, its DCFL.

Context free languages are a proper subset of Recursive Languages. $\therefore$ it is recursive too.

Answer is **option B.**

👍 28 votes -- Amar Vashishth (25.2k points)

5.7.13 Identify Class Language: GATE CSE 2008 | Question: 9 top ☛ https://gateoverflow.in/407

✔ We have algorithms to generate prime numbers $\implies$ we can generate sequence of p for the given language, hence strings as defined by the language definition.

So, by Church Turing Thesis we can say that there exists a Turing Machine which can accept the given language.

Answer is **option D.**

References

👍 29 votes -- Amar Vashishth (25.2k points)

5.7.14 Identify Class Language: GATE CSE 2009 | Question: 40 top ☛ https://gateoverflow.in/1326

✔ **Answer is C.**

$L_1 = \{a^m b^m c a^n b^n \mid m, n \geq 0\}$

It is a CFL. It will generate strings which start with a's followed by equal number of b's and single c followed by a's followed by equal number of b's

i.e., $abcab, aabbcab, aaabbbcaabaabb, \ldots$

$L_2 = \left\{a^i b^j c^k \mid i, j, k \geq 0\right\}$

It is a Regular (and of course CFL). It will generate strings which begin with a's followed by any number of b's followed by any number of c's

So, $L_1 \cap L_2$ will be **all strings which are common in both languages**

So, those will be the strings which begin with a's followed by equal number of b's and ending with c

They would be of the form $a^m b^m c$

Such a language is not regular as it requires equal number of a's and b's. But it is Context Free language as we can make a PDA for it. Also, CFLs are closed under intersection with regular set. So, any CFL intersection a regular set gives a CFL (may or may not be a regular set).

👍 33 votes -- Arunav Khare (3.9k points)

$L_1 \cap L_2 = \{a^m b^m c \mid m \geq 0\}$, which is context free but not regular.

Option **C**.

👍 40 votes -- Arjun Suresh (330k points)

5.7.15 Identify Class Language: GATE CSE 2010 | Question: 40 top ☛ https://gateoverflow.in/2341

✔ Correct Answer: D. All are context free.
$L1 \rightarrow$ Push 0 on stack and when 1 comes, start popping. If stack becomes empty and 1's are remaining start pushing

1. At end of string accept if stack is non- empty.

$L2 \to$ Do the same as for $L1$, but accept if stack is empty at end of string.

$L3 \to$ Do, the same as for $L2$, but for each 0, push two 0's on stack and start the stack with a 0.

$L4 \to$ Do the same as for $L1$, but for each 0, push two 0's on stack

All are in fact DCFL. Pushing two 0's on stack might sound non-trivial but we can do this by pushing one symbol and going to a new state. Then on this new state on empty symbol, push one more symbol on stack and come back.

43 votes -- Arjun Suresh (330k points)

5.7.16 Identify Class Language: GATE CSE 2011 | Question: 26 top

https://gateoverflow.in/2128

✔ Answer is **C.**

L_1 is **RL**

L_2 is **CFL**

L_3 is **CSL**

Turning Machine is powerful Machine it can be used to accept all the languages (RL,CFL,CSL,RE)

32 votes -- Sona Praneeth Akula (3.4k points)

5.7.17 Identify Class Language: GATE CSE 2013 | Question: 32 top

https://gateoverflow.in/1543

✔ L_1 is regular and hence context-free also. Regular expression for L_1 is $0^*1^*0^*$. So, (D) is the false choice.

L_2 is context-free but not regular. We need a stack to count if the number of 0's before and after the 1 (1 may be absent also) are not same. So, $L_1 \cap L_2$ is context-free as regular $\cap$ context-free is context-free. $\to$ (B) is true.

Complement of L_2 is recursive as context-free complement is always recursive (actually even context-sensitive).

45 votes -- Arjun Suresh (330k points)

5.7.18 Identify Class Language: GATE CSE 2014 Set 3 | Question: 36 top

https://gateoverflow.in/2070

✔ Correct Option: **C**.

L_3 is CFL and not DCFL as in no way we can deterministically determine the MIDDLE point of the input string.

37 votes -- Gate Keeda (15.9k points)

5.7.19 Identify Class Language: GATE CSE 2017 Set 1 | Question: 37 top

https://gateoverflow.in/118320

✔ Since while intersection all strings produced by production aSb in G_1 and bSa in G_2 will be 0

So, only common production will be:

$S \to T$

$T \to cT \mid \epsilon$

Which is nothing but c^* hence it is REGULAR and INFINITE

So, option is (**B**).

60 votes -- sriv_shubham (2.8k points)

5.7.20 Identify Class Language: GATE CSE 2017 Set 2 | Question: 40 top

https://gateoverflow.in/118615

✔ L_1 is Csl, L_2 is context free

L_3 is not Context free and L_4 is Dcfl

So, option is **D.**

29 votes -- Swapnil (1.4k points)

5.7.21 Identify Class Language: GATE CSE 2018 | Question: 35 top https://gateoverflow.in/204109

✔ $a)$ $\{a^m b^n c^p d^q \mid m+p=n+q, \text{ where } m,n,p,q \geq 0\}$

Grammar :
$S \longrightarrow aSd \mid ABC \mid \in$
$A \longrightarrow aAb \mid ab \mid \in$
$B \longrightarrow bBc \mid bc \mid \in$
$C \longrightarrow cCd \mid cd \mid \in$

Above Grammar is CFG so it will generate CFL.

$b)$ $\{a^m b^n c^p d^q \mid m=n \text{ and } p=q, \text{ where } m,n,p,q \geq 0\}$

Grammar :
$S \longrightarrow AB \mid \in$
$A \longrightarrow aAb \mid ab$
$B \longrightarrow cBd \mid cd$

Above Grammar is CFG so it will generate CFL.

$c)$ $\{a^m b^n c^p d^q \mid m=n=p \text{ and } p \neq q, \text{ where } m,n,p,q \geq 0\}$
More than one comparison so NOT CFL.

$d)$ $\{a^m b^n c^p d^q \mid mn=p+q, \text{ where } m,n,p,q \geq 0\}$
It is CSL but NOT CFL.

Correct Answer: B

35 votes -- Digvijay (44.9k points)

5.7.22 Identify Class Language: GATE CSE 2020 | Question: 10 top https://gateoverflow.in/333221

✔ $L = \underbrace{\{a^n \mid n \geq 0\}}_{L_1} \cup \underbrace{\{a^n b^n \mid n \geq 0\}}_{L_2}$

Here, L_1 is a regular language having regular expression a^* and L_2 is a DCFL. DCFL is closed under union operator with a regular language as shown below.

$D \cup R = \overline{\overline{D} \cap \overline{R}}$

Since, DCFL and regular sets are closed under complement this gives a DCFL intersection a regular language and DCFL is closed under intersection with regular set. Ref: https://gatecse.in/closure-property-of-language-families/

So, L must be deterministic context-free.

$LL(k)$ parser needs to determine the next grammar production by seeing the next k input symbols. Now, if we have a string like $a^{k+1}\ldots$, the parser has no idea whether to pick the production for a^n or $a^n b^n$. i.e., the input string can be either say a^{k+10} or $a^{k+10}b^{k+10}$, and the $LL(k)$ parser (whatever $LL(k)$ grammar we use for the given language) cannot determine which one it is and so gets stuck. So, the given language is not $LL(k)$ for any k.

Reference: https://gatecse.in/lr-parsing-part-2-language-of-ll-and-lr-grammars/

References

1 votes -- gatecse (62.6k points)

5.7.23 Identify Class Language: GATE CSE 2020 | Question: 32 top

https://gateoverflow.in/333199

✔ Here for L_1 a regular expression exists,

$L_1 = (0+1)^+0(0+1)^+0 + (0+1)^+1(0+1)^+1$

So, L_1 is a Regular Language.

Now, for L_2 a context-free grammer exists which is shown below,

- $S \rightarrow AB \mid BA$
- $A \rightarrow a \mid aAa \mid aAb \mid bAb \mid bAa$
- $B \rightarrow b \mid aBa \mid aBb \mid bBb \mid bBa$

L_2 is the complement of $L = \{ww \mid w \in (a+b)^*\} \cup \{w \mid w \in (a+b)^* \text{ and } |w| \text{ is odd } \}$
Proof : https://cs.stackexchange.com/questions/19151/is-the-complement-of-ww-context-free

So correct option is : A

References

6 votes

-- Aditya Patel (767 points)

5.7.24 Identify Class Language: GATE CSE 2021 Set 2 | Question: 12 top

https://gateoverflow.in/357528

✔ **Correct Answer: B,C,D**

Some insights beforehand:

- **CFLs are Not Closed** under **Intersection**:
  $L_1 = \{a^n b^* c^n\}, L_2 = \{a^n b^n c^*\}$, $L_1 \cap L_2 = \{a^n b^n c^n\}$ not a CFL
- **CFLs are Closed** under **Union:**
  Let both of these CFLs be represented using grammars with start symbols S_1 and S_2 respectively. We define a new symbol S' such that $S' \rightarrow S_1 \mid S_2$. Note: the resulting language may or may not deterministic but will be context-free.
- **CFLs are Not Closed** under **Complementation:**
  $L_1 \cap L_2 = \overline{\overline{L_1} \cup \overline{L_2}}$
  We know that **CFLs** aren't closed under **intersection** but is closed under **union.** So it must be the case that **CFLs** aren't closed under complementation.
  Example: $L = \{a^n b^n c^n\}$ not a CFL(a CSL) but, $\overline{L}$ is and in-fact can be solved using a **NPDA** (construct a NPDA where it checks for either $\text{Number}_a \neq \text{Number}_b$ or $\text{Number}_b \neq \text{Number}_c$ or $\text{Number}_a \neq \text{Number}_c$. But we can't construct one which can check $\text{Number}_a = \text{Number}_b = \text{Number}_c$)
- **Intersection or Union** of a **CFL** and **RL** always gives a **CFL**. So it is closed under the same.

Now,

A. $\overline{L_2}$ is complemented so, needn't be a CFL.
B. $\overline{\overline{L_1} \cap \overline{L_2}} = L_1 \cup L_2$ union of a CFL and RL is a CFL
C. $L_1 \cup (L_2 \cup \overline{L_2}) = L_1 \cup (\Sigma^*) = \Sigma^*$ hence a CFL.
D. $(L_1 \cap L_2) \cup (\overline{L_1} \cap L_2) = L_2$ or it's a union of two CFLs if you don't bother to solve.

3 votes

-- Cringe is my middle name... (817 points)

5.7.25 Identify Class Language: GATE IT 2005 | Question: 4 top

https://gateoverflow.in/3748

✔ Take $L = \Sigma^*$ then $L^c = \varnothing$ and $M^c \cup \varnothing = M^c$

We know that complement of CFL need not be a CFL as CFL is not closed under complement.

So, (A) and (B) are false.

If we take $L = \varnothing$ then $L^c = \Sigma^*$ and $M^c \cup \Sigma^* = \Sigma^*$ which is regular - (C) is also false.

So, answer (**D**)

83 votes -- Arjun Suresh (330k points)

5.7.26 Identify Class Language: GATE IT 2005 | Question: 6 top https://gateoverflow.in/3751

✔ Regular (in fact finite). Since n is finite, we have a finite set of strings satisfying the given conditions. So, we can easily make a finite automata for those strings.

28 votes -- Arjun Suresh (330k points)

5.7.27 Identify Class Language: GATE IT 2008 | Question: 33 top https://gateoverflow.in/3343

✔ Both the languages can be accepted by a **DPDA:**

$L_1 =$ start pushing element X into the stack on input 'a' ... start to pop X on input 'b' ... move to final state when stack empty and input $=$'c'

$L_2 =$ start pushing elements XX into the stack on input 'a' ... start to pop X on input 'b' ... move to final state when stack empty and input $=$ 'epsilon'

So, **(A)** and **(D)** are False.

$L_1 \cup L_2$ is a CFL ... we can build it by having L_1, L_2 and an extra state ... and an 'epsilon' transition to both L_1 and L_2 from that extra state.

So, **(C)** is false.

$L_1 \cap L_2 =$ Phi because we have no string $a^i b^j$ where $i = j$ and $i = 2j$ for $i, j \geq 1$

and clearly L_1 is not a regular language

So, (**B**) is true.

41 votes -- Danish (3.4k points)

5.8 Minimal State Automata (25) top

5.8.1 Minimal State Automata: GATE CSE 1987 | Question: 2j top https://gateoverflow.in/80594

State whether the following statements are TRUE or FALSE:

A minimal DFA that is equivalent to an NDFA with n nodes has always 2^n states.

gate1987 theory-of-computation finite-automata minimal-state-automata

Answer

5.8.2 Minimal State Automata: GATE CSE 1996 | Question: 2.23 top https://gateoverflow.in/2752

Consider the following state table for a sequential machine. The number of states in the minimized machine will be

		Input	
		0	1
Present State	**A**	D,0	B,1
	B	A,0	C,1
	C	A,0	B,1
	D	A,1	C,1
		Next state, Output	

A. 4
B. 3
C. 2
D. 1

gate1996 theory-of-computation normal finite-automata minimal-state-automata

Answer

5.8.3 Minimal State Automata: GATE CSE 1997 | Question: 20 top https://gateoverflow.in/2280

Construct a finite state machine with minimum number of states, accepting all strings over (a, b) such that the number of a's is divisible by two and the number of b's is divisible by three.

gate1997 theory-of-computation finite-automata normal minimal-state-automata descriptive

Answer

5.8.4 Minimal State Automata: GATE CSE 1997 | Question: 70 top https://gateoverflow.in/19700

Following is a state table for time finite state machine.

Present State	Next State Output	
	Input- 0	Input-1
A	B.1	H.1
B	F.1	D.1
C	D.0	E.1
D	C.0	F.1
E	D.1	C.1
F	C.1	C.1
G	C.1	D.1
H	C.0	A.1

A. Find the equivalence partition on the states of the machine.
B. Give the state table for the minimal machine. (Use appropriate names for the equivalent states. For example if states X and Y are equivalent then use XY as the name for the equivalent state in the minimal machine).

gate1997 theory-of-computation minimal-state-automata descriptive

Answer

5.8.5 Minimal State Automata: GATE CSE 1998 | Question: 2.5 top https://gateoverflow.in/1677

Let L be the set of all binary strings whose last two symbols are the same. The number of states in the minimal state deterministic finite state automaton accepting L is

A. 2
B. 5
C. 8
D. 3

gate1998 theory-of-computation finite-automata normal minimal-state-automata

Answer

5.8.6 Minimal State Automata: GATE CSE 1998 | Question: 4 top https://gateoverflow.in/1695

Design a deterministic finite state automaton (using minimum number of states) that recognizes the following language:

$L = \{w \in \{0,1\}^* \mid w$ interpreted as binary number (ignoring the leading zeros) is divisible by five $\}$.

gate1998 theory-of-computation finite-automata normal minimal-state-automata descriptive

Answer

5.8.7 Minimal State Automata: GATE CSE 1999 | Question: 1.4 top https://gateoverflow.in/1458

Consider the regular expression $(0+1)(0+1)\ldots N$ times. The minimum state finite automaton that recognizes the language represented by this regular expression contains

A. n states
B. $n+1$ states
C. $n+2$ states
D. None of the above

gate1999 theory-of-computation finite-automata easy minimal-state-automata

Answer

5.8.8 Minimal State Automata: GATE CSE 2001 | Question: 1.6 top https://gateoverflow.in/699

Given an arbitrary non-deterministic finite automaton (NFA) with N states, the maximum number of states in an equivalent minimized DFA at least

A. N^2
B. 2^N
C. $2N$
D. $N!$

gate2001-cse finite-automata theory-of-computation easy minimal-state-automata

Answer

5.8.9 Minimal State Automata: GATE CSE 2001 | Question: 2.5 top https://gateoverflow.in/723

Consider a DFA over $\Sigma = \{a, b\}$ accepting all strings which have number of a's divisible by 6 and number of b's divisible by 8. What is the minimum number of states that the DFA will have?

A. 8
B. 14
C. 15
D. 48

gate2001-cse theory-of-computation finite-automata minimal-state-automata

Answer

5.8.10 Minimal State Automata: GATE CSE 2002 | Question: 2.13 top https://gateoverflow.in/843

The smallest finite automaton which accepts the language $\{x \mid \text{ length of } x \text{ is divisible by } 3\}$ has

A. 2 states
B. 3 states
C. 4 states
D. 5 states

gate2002-cse theory-of-computation normal finite-automata minimal-state-automata

Answer

5.8.11 Minimal State Automata: GATE CSE 2006 | Question: 34 top https://gateoverflow.in/1291

Consider the regular language $L = (111 + 11111)^*.$ The minimum number of states in any DFA accepting this languages is:

A. 3
B. 5
C. 8
D. 9

gate2006-cse theory-of-computation finite-automata normal minimal-state-automata

Answer

5.8.12 Minimal State Automata: GATE CSE 2007 | Question: 29 top https://gateoverflow.in/1227

A minimum state deterministic finite automaton accepting the language

$L = \{w \mid w \in \{0, 1\}^*, \text{ number of 0s and 1s in } w \text{ are divisible by } 3 \text{ and } 5\text{, respectively }\}$ has

A. 15 states
B. 11 states

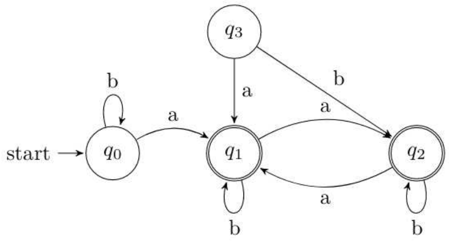
C. 10 states
D. 9 states

gate2007-cse theory-of-computation finite-automata normal minimal-state-automata

Answer

5.8.13 Minimal State Automata: GATE CSE 2007 | Question: 75 top https://gateoverflow.in/43514

Consider the following Finite State Automaton:



The minimum state automaton equivalent to the above FSA has the following number of states:

A. 1
B. 2
C. 3
D. 4

normal gate2007-cse theory-of-computation finite-automata minimal-state-automata

Answer

5.8.14 Minimal State Automata: GATE CSE 2010 | Question: 41 top https://gateoverflow.in/2342

Let w be any string of length n in $\{0,1\}^*$. Let L be the set of all substrings of w. What is the minimum number of states in non-deterministic finite automation that accepts L?

A. $n-1$
B. n
C. $n+1$
D. 2^{n-1}

gate2010-cse theory-of-computation finite-automata normal minimal-state-automata

Answer

5.8.15 Minimal State Automata: GATE CSE 2011 | Question: 42 top https://gateoverflow.in/2144

Definition of a language L with alphabet $\{a\}$ is given as following.

$$L = \{a^{nk} \mid k > 0, \ \ and \ \ n \text{ is a positive integer constant}\}$$

What is the minimum number of states needed in a DFA to recognize L?

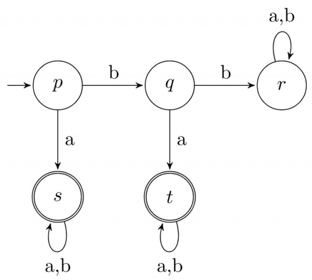
A. $k+1$
B. $n+1$
C. 2^{n+1}
D. 2^{k+1}

gate2011-cse theory-of-computation finite-automata normal minimal-state-automata

Answer

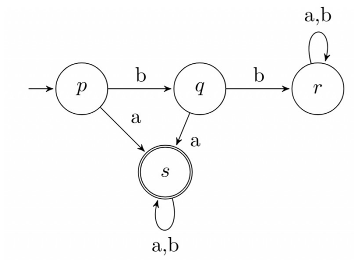
5.8.16 Minimal State Automata: GATE CSE 2011 | Question: 45 top https://gateoverflow.in/2147

A deterministic finite automaton (DFA) D with alphabet $\Sigma = \{a,b\}$ is given below.

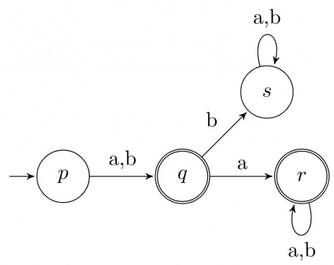


Which of the following finite state machines is a valid minimal DFA which accepts the same languages as D?

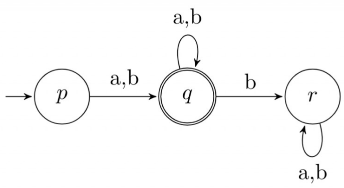
A.



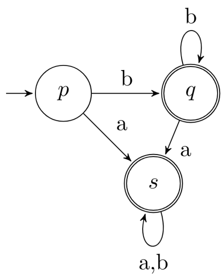
B.



C.



D.

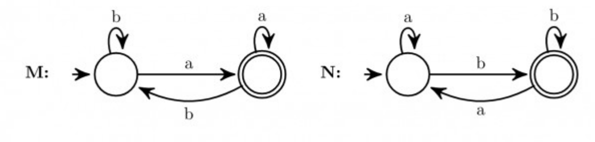


gate2011-cse theory-of-computation finite-automata easy minimal-state-automata

Answer

5.8.17 Minimal State Automata: GATE CSE 2015 Set 1 | Question: 52 top

https://gateoverflow.in/8362



Consider the DFAs M and N given above. The number of states in a minimal DFA that accept the language $L(M) \cap L(N)$ is_____________.

gate2015-cse-set1 theory-of-computation finite-automata easy numerical-answers minimal-state-automata

Answer

5.8.18 Minimal State Automata: GATE CSE 2015 Set 2 | Question: 53 top https://gateoverflow.in/8256

The number of states in the minimal deterministic finite automaton corresponding to the regular expression $(0+1)^*(10)$ is _____.

gate2015-cse-set2 theory-of-computation finite-automata normal numerical-answers minimal-state-automata

Answer

5.8.19 Minimal State Automata: GATE CSE 2015 Set 3 | Question: 18 top https://gateoverflow.in/8415

Let L be the language represented by the regular expression $\Sigma^*0011\Sigma^*$ where $\Sigma = \{0, 1\}$. What is the minimum number of states in a DFA that recognizes $\bar{L}$ (complement of L)?

A. 4
B. 5
C. 6
D. 8

gate2015-cse-set3 theory-of-computation finite-automata normal minimal-state-automata

Answer

5.8.20 Minimal State Automata: GATE CSE 2016 Set 2 | Question: 16 top https://gateoverflow.in/39562

The number of states in the minimum sized DFA that accepts the language defined by the regular expression.

$$(0+1)^*(0+1)(0+1)^*$$

is ________.

gate2016-cse-set2 theory-of-computation finite-automata normal numerical-answers minimal-state-automata

Answer

5.8.21 Minimal State Automata: GATE CSE 2017 Set 1 | Question: 22 top https://gateoverflow.in/118302

Consider the language L given by the regular expression $(a+b)^*b(a+b)$ over the alphabet $\{a, b\}$. The smallest number of states needed in a deterministic finite-state automaton (DFA) accepting L is ___________ .

gate2017-cse-set1 theory-of-computation finite-automata numerical-answers minimal-state-automata

Answer

5.8.22 Minimal State Automata: GATE CSE 2017 Set 2 | Question: 25 top https://gateoverflow.in/118160

The minimum possible number of states of a deterministic finite automaton that accepts the regular language $L = \{ w_1aw_2 \mid w_1, w_2 \in \{a,b\}^*, |w_1| = 2, |w_2| \geq 3 \}$ is ______________ .

theory-of-computation gate2017-cse-set2 finite-automata numerical-answers minimal-state-automata

Answer

5.8.23 Minimal State Automata: GATE CSE 2018 | Question: 6 top https://gateoverflow.in/204080

Let N be an NFA with n states. Let k be the number of states of a minimal DFA which is equivalent to N. Which one of the following is necessarily true?

A. $k \geq 2^n$
B. $k \geq n$
C. $k \leq n^2$
D. $k \leq 2^n$

gate2018-cse theory-of-computation minimal-state-automata normal

Answer

5.8.24 Minimal State Automata: GATE CSE 2019 | Question: 48 top https://gateoverflow.in/302800

Let Σ be the set of all bijections from $\{1, \dots, 5\}$ to $\{1, \dots, 5\}$, where id denotes the identity function, i.e. $id(j) = j, \forall j$. Let $\circ$ denote composition on functions. For a string $x = x_1 x_2 \dots x_n \in \Sigma^n, n \geq 0$, let $\pi(x) = x_1 \circ x_2 \circ \dots \circ x_n$. Consider the language $L = \{x \in \Sigma^* \mid \pi(x) = id\}$. The minimum number of states in any DFA accepting L is _______

gate2019-cse numerical-answers theory-of-computation finite-automata minimal-state-automata difficult

Answer

5.8.25 Minimal State Automata: GATE IT 2008 | Question: 6 top https://gateoverflow.in/3266

Let N be an NFA with n states and let M be the minimized DFA with m states recognizing the same language. Which of the following in NECESSARILY true?

A. $m \leq 2^n$
B. $n \leq m$
C. M has one accept state
D. $m = 2^n$

gate2008-it theory-of-computation finite-automata normal minimal-state-automata

Answer

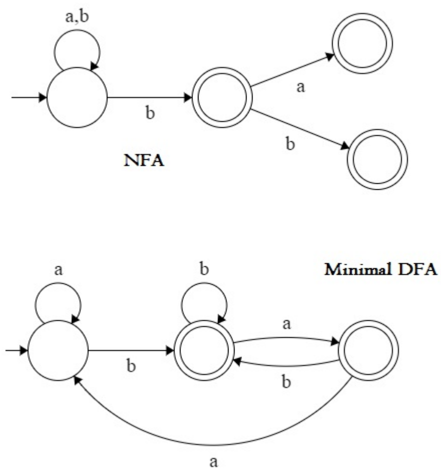
Answers: Minimal State Automata

5.8.1 Minimal State Automata: GATE CSE 1987 | Question: 2j top https://gateoverflow.in/80594

✔ No a minimal DFA that is equiavlent to an NDFA with n nodes has always 2^n states.

Correct statement is A minimal DFA that is equivalent to an NDFA with n nodes has atmost 2^n states.

Example :

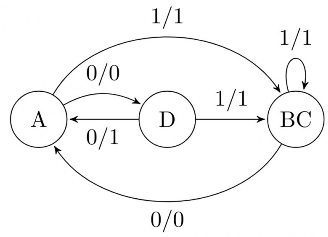


👍 28 votes

-- Prashant Singh (47.1k points)

5.8.2 Minimal State Automata: GATE CSE 1996 | Question: 2.23 top https://gateoverflow.in/2752

✔ 3 states are required in the minimized machine. States B and C can be combined as follows:



👍 33 votes

-- Arjun Suresh (330k points)

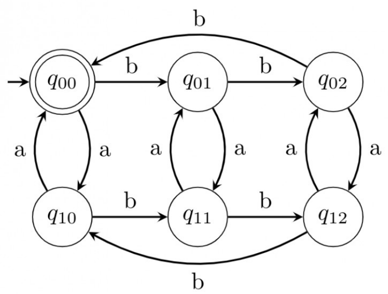
5.8.3 Minimal State Automata: GATE CSE 1997 | Question: 20 top https://gateoverflow.in/2280

✔ A state q_{xy} means $n_a \mod 2 = x$, $n_b \mod 3 = y$

q_{00} means $n_a \mod 2 = 0, n_b \mod 3 = 0$ [no of a's is divisible by 2 and no of b's is divisible by 3]

$q_{00} \times a \rightarrow q_{10}$

$q_{00} \times b \rightarrow q_{01}$ and so on.



👍 23 votes

-- Praveen Saini (41.9k points)

5.8.4 Minimal State Automata: GATE CSE 1997 | Question: 70 top https://gateoverflow.in/19700

✔ As nothing is mentioned in question, assuming that the first state itself is the start state i.e. State A so, State G is not reachable from the start and hence is removed before applying the minimization algorithm.

- $P0 \rightarrow [ABEFG][CDH] \rightarrow [ABEF][CDH]$
- $P1 \rightarrow [AB][EF][CDH]$
- $P2 \rightarrow [A][B][EF][CD][H]$
- $P3 \rightarrow [A][B][EF][CD][H]$
  $//P2 == P3$ So, stop

State table for the minimal machine ::

Present State	input $= 0$	input $= 1$
A	B.1	H.1
B	EF.1	CD.1
CD	CD.0	EF.1
EF	CD.1	CD.1
H	CD.0	A.1

👍 21 votes

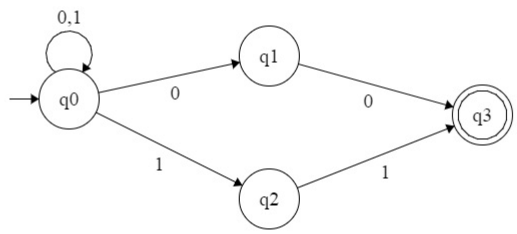
-- Vidhi Sethi (8.3k points)

5.8.5 Minimal State Automata: GATE CSE 1998 | Question: 2.5 top https://gateoverflow.in/1677

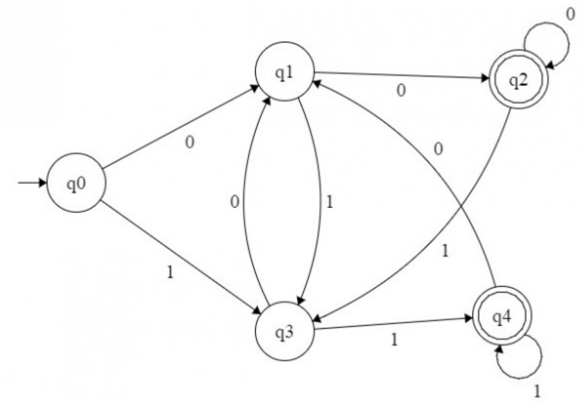
✔ Binary strings whose last two symbols are same .

Regular expression $= (0 + 1)^*(00 + 11)$

Having NFA



Equivalent DFA



Total no of states $= 5$

Correct Answer: B

48 votes -- Praveen Saini (41.9k points)

5.8.6 Minimal State Automata: GATE CSE 1998 | Question: 4 top https://gateoverflow.in/1695

✔ Suppose we have decimal no 3 after that we get a symbol 2 . it becomes 32 as $3x10 + 2 = 32$

In binary if we have 10 (i.e 2 in decimal say old no) and after that we get symbol 1 it become 101(i.e 5 in decimal say new no)

2 (old no.) $\times 2$(base) $+1$(input symbol) $= 5$ (new no.)

Now in the given problem, binary no is divisible by 5 , i.e $0, 101, 1010, 1111.....$

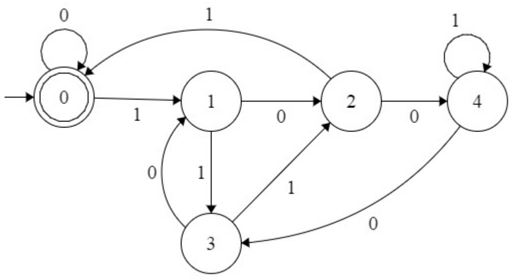
We need 5 states in DFA , $0, 1, 2, 3$ and 4 .Each state represent remainder that comes when we divide no by 5.

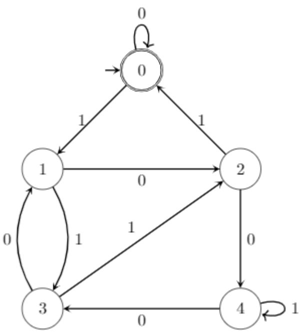
Input symbol $= \{0, 1\}$

We get the transition:

[Old state $\times$ base $+$ input symbol] mod $5 =$ New state , where base is 2

$[0 \times 2 + 0] mod\ 5 = 0$	$[1 \times 2 + 0] mod\ 5 = 2$	$[2 \times 2 + 0] mod\ 5 = 4$
$[0 \times 2 + 1] mod\ 5 = 1$	$[1 \times 2 + 1] mod\ 5 = 3$	$[2 \times 2 + 1] mod\ 5 = 0$
$[3 \times 2 + 0] mod\ 5 = 1$	$[4 \times 2 + 0] mod\ 5 = 3$	
$[3 \times 2 + 1] mod\ 5 = 2$	$[4 \times 2 + 1] mod\ 5 = 4$	



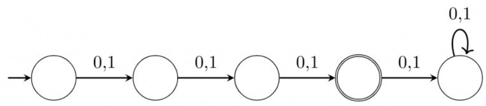


👍 44 votes -- Praveen Saini (41.9k points)

5.8.7 Minimal State Automata: GATE CSE 1999 | Question: 1.4 top https://gateoverflow.in/1458

✔ As far as for above problem say regular expression for $(0+1)(0+1)\dots 3$ times $=(0+1)(0+1)(0+1)$

Having **DFA:**



So, for regular expression $(0+1)(0+1)\dots N$ times we have $\mathbf{N+2}$ states in DFA

$N+1$ states in NFA (we can remove dead state)

When question is about minimum state finite automata (and nothing is mentioned NFA/DFA) then which ever having minimum number must be taken which here is $N+1$ states.

Correct Answer: B

👍 60 votes -- Praveen Saini (41.9k points)

5.8.8 Minimal State Automata: GATE CSE 2001 | Question: 1.6 top https://gateoverflow.in/699

✔ Correct Option: **B**

In **DFA** any subset of the N states (for N element set 2^N subsets possible) can become a new state and they can remain even when the DFA is minimized. So, maximum we can get 2^N states for the minimized **DFA**. (at least in question must be a typo for at most).

👍 33 votes -- jayendra (6.7k points)

5.8.9 Minimal State Automata: GATE CSE 2001 | Question: 2.5 top https://gateoverflow.in/723

✔ Correct Option: **D**

It can be proved using Myhill Nerode theorem. We need a separate state for $\#a \; mod \; 6 = 0..5$ and $\#b \; mod \; 8 = 0..7$. Each combination of them must also be a new state giving $6*8=48$ minimum states required in the DFA.

Reading Myhill-Nerode theorem might be confusing though it is actually very simple.

http://courses.cs.washington.edu/courses/cse322/05wi/handouts/MyhillNerode.pdf

References

👍 38 votes -- Arjun Suresh (330k points)

5.8.10 Minimal State Automata: GATE CSE 2002 | Question: 2.13 top https://gateoverflow.in/843

✔ Correct Option: **B**

It is 3 states as we need a state each for length mod $3=0,1$ and 2.

👍 26 votes -- priya023 (137 points)

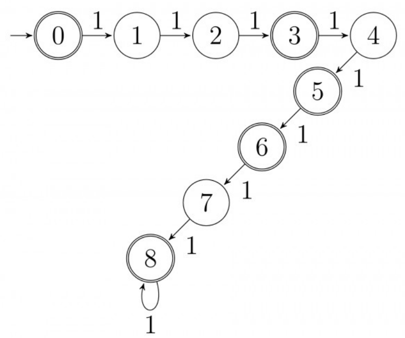
5.8.11 Minimal State Automata: GATE CSE 2006 | Question: 34 top ☛ https://gateoverflow.in/1291

✔ Given language $L = (111 + 11111)^*$
Strings , that belongs to the language

$L = \{\epsilon, 111, 11111, 111111, 11111111, 111111111, 1111111111, \dots$

Due to concatenation with $(111)^*$ if we have three consecutive string lengths in L, then all higher string lengths will be in L.

We have strings of length $8, 9, 10$ in L and so all higher length strings are also in L.
So, required **DFA** will be:



So, there are 5 states are final states and 4 states are non-final states ,total number of states are 9 states .
Hence, option **D** is true.

👍 76 votes -- Mithlesh Upadhyay (4.3k points)

5.8.12 Minimal State Automata: GATE CSE 2007 | Question: 29 top ☛ https://gateoverflow.in/1227

✔ Answer will be (**A**) 15 states.

We need a separate state for $\#0 = 0, 1, 2$ and $\#1 = 0, 1, 2, 3, 4$ giving total minimum number of states $= 3 * 5 = 15$.

This is a direct consequence of Myhill-Nerode theorem.

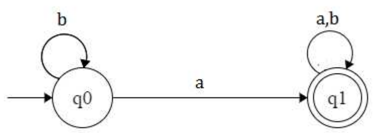
http://courses.cs.washington.edu/courses/cse322/05wi/handouts/MyhillNerode.pdf

References

👍 28 votes -- Arjun Suresh (330k points)

5.8.13 Minimal State Automata: GATE CSE 2007 | Question: 75 top ☛ https://gateoverflow.in/43514

✔ Answer will be **2** state.



👍 29 votes -- Manoj Kumar (26.7k points)

5.8.14 Minimal State Automata: GATE CSE 2010 | Question: 41 top ☛ https://gateoverflow.in/2342

✔ We need a state for counting the length. So, for length n we need $n + 1$ states (one for length zero). We don't need a reject state for larger strings as we have NFA and not DFA. So, totally $n + 1$ states are required. (For DFA it would be $n + 2$).

Correct Answer: C

👍 67 votes -- Arjun Suresh (330k points)

5.8.15 Minimal State Automata: GATE CSE 2011 | Question: 42 top https://gateoverflow.in/2144

✔ (**B**) $n+1$

We need a state for strings of length $0, 1, 2, \dots n$ (and their respective multiples with k). Each of these set of strings form an equivalence class as per Myhill-Nerode relation and hence needs a separate state in min-DFA.

Myhill-Nerode Class 1	Myhill-Nerode Class 2	Myhill-Nerode Class n	Myhill-Nerode Class n+1
ϵ	a, #a=n+1, #a=2n+1, ...	#a=n-1, #a=2n-1, #a=3n-1, ...	#a=n, #a=2n, #a=3n, ...

One thing to notice here is $k > 0$. Because of this we are not able to combine Class 1 and Class $n+1$. Had it been $k \geq 0$, we would have had only n equivalent classes and equivalently n states in the minimal DFA.

40 votes -- Arjun Suresh (330k points)

5.8.16 Minimal State Automata: GATE CSE 2011 | Question: 45 top https://gateoverflow.in/2147

✔ Correct Option: **A**

In **(B)** and **(C)** when the first letter of input is 'b' we reach final state, while in the given **DFA** first letter 'b' is not a final state. So, **(B)** and **(C)** are not accepting same language as the given **DFA.**

In **(D)** we can reach final state when the last letter is 'a', whatever be the previous transitions. But in the given **DFA**, when the first 2 letters are 'b' we can never reach the final state. So, **(D)** is also accepting a different language than the given **DFA.**

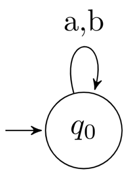
37 votes -- Arjun Suresh (330k points)

5.8.17 Minimal State Automata: GATE CSE 2015 Set 1 | Question: 52 top https://gateoverflow.in/8362

✔ $L(M) = (a+b)^*\, a = \{a, aa, ba, aaa, aba, bba, \dots\}$

$L(N) = (a+b)^*\, b = \{b, ab, bb, aab, abb, bbb, \dots\}$

So, $L(M) \cap L(N) = \{\}$. So, in the minimal DFA, we just have 1 start state with all transitions going to it self and no final state.



103 votes -- Arjun Suresh (330k points)

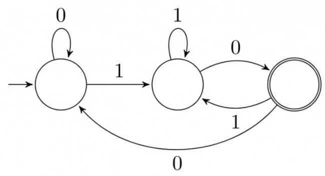
5.8.18 Minimal State Automata: GATE CSE 2015 Set 2 | Question: 53 top https://gateoverflow.in/8256

✔ All strings ending with 10. So, we need 3 states.

1. From first state on 1, we go to second state.
2. From second state on 0 we go to third state.
3. From third state on 0 we go to first state and on 1 we go to second state.

Only third state is final.

`L = (0+1)*10` Minimal DFA will be as follows:



45 votes -- Arjun Suresh (330k points)

5.8.19 Minimal State Automata: GATE CSE 2015 Set 3 | Question: 18 top https://gateoverflow.in/8415

✔ First we can draw dfa for L which has 5 states after that for L compliment we will convert all final to non final and all non final to final so, total states is 5.

Anwer is option **B.**

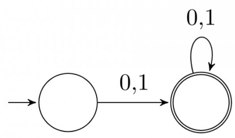
50 votes -- Anoop Sonkar (4.1k points)

5.8.20 Minimal State Automata: GATE CSE 2016 Set 2 | Question: 16 top https://gateoverflow.in/39562

✔ All strings over $\{0, 1\}$ having length ≥ 1

$(0+1)^*(0+1)(0+1)^* = (0+1)(0+1)^* = (0+1)^*(0+1) = (0+1)^+$

Having **DFA:**

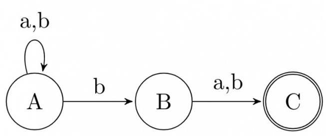


Number of states in minimal **DFA** $= 2$.

80 votes -- Praveen Saini (41.9k points)

5.8.21 Minimal State Automata: GATE CSE 2017 Set 1 | Question: 22 top https://gateoverflow.in/118302

✔



NFA

NFA to DFA Conversion:

In the State Transition Table below we can see **4 states** $(\text{A}, \text{AB}, {}^*\text{AC}, {}^*\text{ABC} -- \text{the } {}^* \text{ ones being FINAL})$ are there but before confirming the ans **lets try to minimize.** A, AB cannot be minimized further because A goes to non-final state on a where as AB goes to a final state. Similarly ${}^*\text{AC}$ goes to a non-final state on **a** whereas ${}^*\text{ABC}$ goes to a final state. Hence, none of the states can be merged.

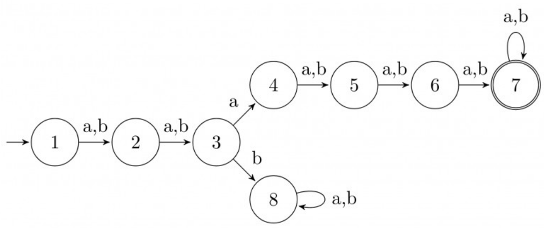
δ	a	b
A	A	AB
AB	*AC	*ABC
*AC	A	AB
*ABC	*AC	*ABC

Answer: 4.

53 votes -- Ahwan Mishra (10.2k points)

5.8.22 Minimal State Automata: GATE CSE 2017 Set 2 | Question: 25 top https://gateoverflow.in/118160

✔ Answer is **8** states included one trap state **(8)** and final state **(7)**.



3^{rd} symbol from the start is an **a**

👍 71 votes

-- Prashant Singh (47.1k points)

5.8.23 Minimal State Automata: GATE CSE 2018 | Question: 6 top

☛ https://gateoverflow.in/204080

✔ Number of states in minimal DFA - k must be $\leq 2^n$ as each state corresponds to a subset of states of the corresponding NFA.

D is answer.

Option B is not always TRUE as the NFA N can have epsilon moves and hence can have more number of states than the minimal DFA.

👍 24 votes

-- Prashant Singh (47.1k points)

5.8.24 Minimal State Automata: GATE CSE 2019 | Question: 48 top

☛ https://gateoverflow.in/302800

✔ Answer -120

I am rewriting question in (hopefully) simple way-

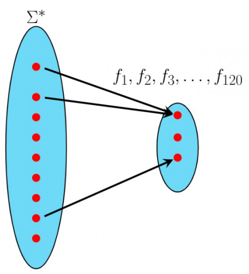
There are $5! = 120$ bijection functions possible. let $f_1, f_2, \ldots f_{120}$ be functions. we can represent every function f_i with some string x_i.

Now we take the composition of functions f_i and f_j then the corresponding string is $x_i x_j$ (Composition of two functions in function space is the concatenation of corresponding strings in string space).

eg: Composite function $f_i \circ f_j \circ f_k$ has representation $x_i x_j x_k$ and so on.

A function can have multiple string representations, suppose $f_i = f_j \circ f_k$ then f_i has x_i and $x_j x_k$ both representations. And further, suppose $f_i = f_p \circ f_q \circ f_r$ then $x_p x_q x_r$ also maps to f_i.

There is a given function called π which does the same job. i.e. π maps a string to the corresponding function. (It is obvious to see that π is many to one function.)



We need to construct a DFA, which accepts a set of all strings whose function representation behaves like Identity function.

$x_1 x_2 \ldots x_n$ is accepted If and Only If $f_1 \circ f_2 \circ \cdots \circ f_n$ is identity function.

In other words, Design a DFA for $L = \pi^{-1}(id)$.

Solution:

First interesting and important point to note is, $\pi(\epsilon) = id$.

Quick check for above statement: Suppose it is not the case and $\pi(\epsilon)$ is some nonidentity function f. Then consider a string $x_1 x_2 x_3$ which maps to a function f_{123} (where $f_{123} = f_1 \circ f_2 \circ f_3$) i.e. $\pi(x_1 x_2 x_3) = f_1 \circ f_2 \circ f_3 = f_{123}$. Now we can always append epsilon to a string $x_1 x_2 x_3 = x_1 x_2 x_3 \epsilon$. $\pi(x_1 x_2 x_3 \epsilon) = f_1 \circ f_2 \circ f_3 \circ f = f_{123} \circ f \neq f_{123}$.

This concludes that $\pi(\epsilon) = id$, therefore, ϵ must be accepted by our DFA.

Now we can construct a DFA over alphabet $\Sigma = \{x_1, x_2, \ldots, x_{120}\}$ which has 120 states corresponding to 120 functions.

Initial state and final state will be the same which corresponds to the Identity function.

For any string $x_i x_j$, DFA will move to state corresponding to function $f_i \circ f_j$ and let $f_k = f_i \circ f_j$. Therefore on string $x_i x_j$, DFA will be in state corresponding to function f_k.

(I have written $f_k = f_i \circ f_j$, where f_k is one of the function in $f_1, f_2, \ldots, f_{120}$, because this set of bijective functions $\{f_1, f_2, \ldots, f_{120}\}$ is closed under composition).

Formally, we have DFA $M = (\Sigma, Q, q_0, \delta, F)$ for $L = \pi^{-1}(id)$.

where, $\Sigma = \{x_1, x_2, \ldots, x_{120}\}$

$Q = \{f_1, f_2, \ldots, f_{120}\}$ (let name of i^{th} state is f_i)

$q_0 = f_{id}$ (f_{id} is identity function)

$\delta(\mathbf{f_i}, \mathbf{a}) = \mathbf{f_i} \circ \pi(\mathbf{a}), \ \forall a \in \Sigma$

$F = \{f_{id}\}$

30 votes -- Sachin Mittal (15.8k points)

Number of bijective functions from $\{1, \ldots, 5\}$ to $\{1, \ldots, 5\} = 5! = 120$.

Now, our alphabet set Σ is having 120 elements -- say for example first 120 ASCII characters representing each of these 120 distinct functions.

Out of these 120 bijective functions there is exactly one identity function - say it is denoted by the ASCII character I in our example alphabet set.

Now, say we make an NFA with 120 states such that from the initial state we move to state represented by function f_i for the symbol corresponding to f_i. i.e., in our ASCII set, for symbol $'A'$ we move to state A, for symbol $'B'$ we move to state B etc. and for symbol I we say in same state. In detail,

- For first symbol of input string, we say in start state if the symbol is I
- For any other 119 symbols possible we move to the corresponding state for that symbol
- Now say for symbol A we moved to state A and the second symbol is K where the function represented by K is the inverse of the function represented by A. In this case we move back to the start state
- We can assume each of the state represents a permutation of $1, 2, \ldots 5$
- From any state, represented by a permutation of $1, 2, \ldots 5$ say s, for the next symbol b we move to the state given by f_b applied on s.
- When the string ends, if we happens to be in start state, or equivalently we simulated an Identity function, then we accept. Else reject.

If we see the above NFA is actually a DFA and we cannot minimize it. So, we will need minimum 120 states to recognize L.

PS: The mapping of the 120 functions to the corresponding symbols is assumed in the question

12 votes -- Arjun Suresh (330k points)

5.8.25 Minimal State Automata: GATE IT 2008 | Question: 6 top https://gateoverflow.in/3266

✔ A state in a DFA will be a subset of the set of states of the equivalent NFA. So, the maximum number of states in the equivalent DFA of an NFA, will be 2^n, where n is the number of states in NFA, as a set with n items has maximum 2^n subsets.

So, answer here is (**A**).

32 votes -- Arjun Suresh (330k points)

5.9 Non Determinism (7) top

5.9.1 Non Determinism: GATE CSE 1992 | Question: 02,xx top https://gateoverflow.in/577

In which of the cases stated below is the following statement true?
"For every non-deterministic machine M_1 there exists an equivalent deterministic machine M_2 recognizing the same language".

A. M_1 is non-deterministic finite automaton.
B. M_1 is non-deterministic PDA.
C. M_1 is a non-deterministic Turing machine.
D. For no machines M_1 and M_2, the above statement true.

gate1992 theory-of-computation easy non-determinism multiple-selects

Answer

5.9.2 Non Determinism: GATE CSE 1994 | Question: 1.16 top https://gateoverflow.in/2459

Which of the following conversions is not possible (algorithmically)?

A. Regular grammar to context free grammar
B. Non-deterministic FSA to deterministic FSA
C. Non-deterministic PDA to deterministic PDA
D. Non-deterministic Turing machine to deterministic Turing machine

gate1994 theory-of-computation easy non-determinism

Answer

5.9.3 Non Determinism: GATE CSE 1998 | Question: 1.11 top https://gateoverflow.in/1648

Regarding the power of recognition of languages, which of the following statements is false?

A. The non-deterministic finite-state automata are equivalent to deterministic finite-state automata.
B. Non-deterministic Push-down automata are equivalent to deterministic Push-down automata.
C. Non-deterministic Turing machines are equivalent to deterministic Turing machines.
D. Multi-tape Turing machines are available are equivalent to Single-tape Turing machines.

gate1998 theory-of-computation easy non-determinism

Answer

5.9.4 Non Determinism: GATE CSE 2005 | Question: 54 top https://gateoverflow.in/1377

Let N_f and N_p denote the classes of languages accepted by non-deterministic finite automata and non-deterministic push-down automata, respectively. Let D_f and D_p denote the classes of languages accepted by deterministic finite automata and deterministic push-down automata respectively. Which one of the following is TRUE?

A. $D_f \subset N_f$ and $D_p \subset N_p$
B. $D_f \subset N_f$ and $D_p = N_p$
C. $D_f = N_f$ and $D_p = N_p$
D. $D_f = N_f$ and $D_p \subset N_p$

gate2005-cse theory-of-computation easy non-determinism

Answer

5.9.5 Non Determinism: GATE CSE 2009 | Question: 16, ISRO2017-12 top https://gateoverflow.in/1308

Which one of the following is FALSE?

A. There is a unique minimal DFA for every regular language
B. Every NFA can be converted to an equivalent PDA.
C. Complement of every context-free language is recursive.
D. Every nondeterministic PDA can be converted to an equivalent deterministic PDA.

gate2009-cse theory-of-computation easy isro2017 non-determinism

Answer

5.9.6 Non Determinism: GATE CSE 2011 | Question: 8 top https://gateoverflow.in/2110

Which of the following pairs have **DIFFERENT** expressive power?

A. Deterministic finite automata (DFA) and Non-deterministic finite automata (NFA)
B. Deterministic push down automata (DPDA) and Non-deterministic push down automata (NPDA)
C. Deterministic single tape Turing machine and Non-deterministic single tape Turing machine
D. Single tape Turing machine and multi-tape Turing machine

gate2011-cse theory-of-computation easy non-determinism

Answer

5.9.7 Non Determinism: GATE IT 2004 | Question: 9 top https://gateoverflow.in/3650

Which one of the following statements is FALSE?

A. There exist context-free languages such that all the context-free grammars generating them are ambiguous
B. An unambiguous context-free grammar always has a unique parse tree for each string of the language generated by it
C. Both deterministic and non-deterministic pushdown automata always accept the same set of languages
D. A finite set of string from some alphabet is always a regular language

gate2004-it theory-of-computation easy non-determinism

Answer

Answers: Non Determinism

5.9.1 Non Determinism: GATE CSE 1992 | Question: 02,xx top https://gateoverflow.in/577

✔ Answer: **A** and **C**.

- For every NFA there exists a DFA.
- For every NPDA there does not exist a deterministic PDA.
- Every nondeterministic Turing machine has an equivalent deterministic Turing Machine.

28 votes -- Rajarshi Sarkar (27.8k points)

5.9.2 Non Determinism: GATE CSE 1994 | Question: 1.16 top https://gateoverflow.in/2459

✔ Correct Option: **C**

Because if that would have been possible then NPDA and DPDA must had same powers, which is not the case. You can take example of NFA and DFA. Both are convertible to each other and hence share the same power.

24 votes -- Gate Keeda (15.9k points)

5.9.3 Non Determinism: GATE CSE 1998 | Question: 1.11 top https://gateoverflow.in/1648

✔

A. True. Proof - Subset Construction Procedure.
B. False. Conversion from **NPDA** To **DPDA** is not always possible like in the case of ww_R.
C. True. For any non-deterministic TM we can construct an equivalent (accepting the same language) deterministic TM.
D. True. For an $n - \texttt{tape}$ TM, we can always construct an equivalent single tape TM.

Answer: B

25 votes -- Akash Kanase (36k points)

5.9.4 Non Determinism: GATE CSE 2005 | Question: 54 top https://gateoverflow.in/1377

✔ Correct Option: **D**

NFA and DFA both have equivalent power.(every nfa can be converted into equivalent DFA) but NPDA can accept more languages than DPDA.

24 votes -- shreya ghosh (2.8k points)

5.9.5 Non Determinism: GATE CSE 2009 | Question: 16, ISRO2017-12 top

https://gateoverflow.in/1308

✔ Correct Option: **D**

NDPA is more powerful than **DPDA**, so they are not equivalent. Actually, **DPDA** is a proper subset of **NDPA.**

C is TRUE as **CFL** is a proper subset of recursive languages and recursive languages are closed under complement.

47 votes -- Bhagirathi Nayak (11.7k points)

5.9.6 Non Determinism: GATE CSE 2011 | Question: 8 top

https://gateoverflow.in/2110

✔ Expressing power of any machine can be defined as the maximum number of languages it can accept..if machine M_1 can accept more languages then M_2 then we can say that expressing power of M_1 is greater then M_2.

A. Languages accepted by NFA,will also be accepted by DFA because we can make DFA corresponding to NFA. So their expressing power is same.
B. In this case languages accepted by NPDA is more then DPDA, so expressing power of NPDA is more then DPDA
C. Both deterministic and non deterministic turing can accept same language.so there expressing power is same.
D. In turing machine if we increase the number of tape then also language accepted by that machine is same as single tape turing machine.so there expressing power is same.

Answer is **B.**

30 votes -- neha pawar (3.3k points)

5.9.7 Non Determinism: GATE IT 2004 | Question: 9 top

https://gateoverflow.in/3650

✔

A. This is true for inherently ambiguous language
B. Always correct, that's why called unambiguous
C. NPDA is a super set of DPDA, hence it's **FALSE**
D. Finite language is always regular

30 votes -- Manu Thakur (34.1k points)

5.10 P Np Npc Nph (5) top

5.10.1 P Np Npc Nph: GATE CSE 2005 | Question: 58 top

https://gateoverflow.in/1381

Consider the following two problems on undirected graphs:

- α: Given $G(V, E)$, does G have an independent set of size |V| - 4?
- β: Given $G(V, E)$, does G have an independent set of size 5?

Which one of the following is TRUE?

A. α is in P and β is NP-complete
B. α is NP-complete and β is in P
C. Both α and β are NP-complete
D. Both α and β are in P

gate2005-cse theory-of-computation p-np-npc-nph normal

Answer

5.10.2 P Np Npc Nph: GATE CSE 2006 | Question: 31 top

https://gateoverflow.in/994

Let SHAM_3 be the problem of finding a Hamiltonian cycle in a graph $G = (V, E)$ with $|V|$ divisible by 3 and DHAM_3 be the problem of determining if a Hamiltonian cycle exists in such graphs. Which one of the following is true?

A. Both DHAM_3 and SHAM_3 are NP-hard
B. SHAM_3 is NP-hard, but DHAM_3 is not
C. DHAM_3 is NP-hard, but SHAM_3 is not
D. Neither DHAM_3 nor SHAM_3 is NP-hard

gate2006-cse theory-of-computation p-np-npc-nph normal

Answer

5.10.3 P Np Npc Nph: GATE CSE 2009 | Question: 14 top https://gateoverflow.in/1306

Let π_A be a problem that belongs to the class NP. Then which one of the following is TRUE?

A. There is no polynomial time algorithm for π_A.
B. If π_A can be solved deterministically in polynomial time, then P = NP.
C. If π_A is NP-hard, then it is NP-complete.
D. π_A may be undecidable.

gate2009-cse theory-of-computation p-np-npc-nph

Answer

5.10.4 P Np Npc Nph: GATE CSE 2012 | Question: 4 top https://gateoverflow.in/36

Assuming $P \neq NP$, which of the following is **TRUE**?

A. $NP-\ complete = NP$
B. $NP-complete \cap P = \phi$
C. $NP-hard = NP$
D. $P = NP-complete$

gate2012-cse theory-of-computation p-np-npc-nph

Answer

5.10.5 P Np Npc Nph: GATE CSE 2013 | Question: 18 top https://gateoverflow.in/1440

Which of the following statements are **TRUE**?

1. The problem of determining whether there exists a cycle in an undirected graph is in P.
2. The problem of determining whether there exists a cycle in an undirected graph is in NP.
3. If a problem A is $NP-Complete$, there exists a non-deterministic polynomial time algorithm to solve A

A. 1, 2 and 3
B. 1 and 2 only
C. 2 and 3 only
D. 1 and 3 only

gate2013-cse theory-of-computation p-np-npc-nph normal

Answer

Answers: P Np Npc Nph

5.10.1 P Np Npc Nph: GATE CSE 2005 | Question: 58 top https://gateoverflow.in/1381

✔ Independent Set- a set of vertices in a graph no two of which are adjacent.

Maximal Independent set problem - Given a graph G, find the size of the maximal independent set in it. This problem in NP-hard.

Independent set decision problem - Given a graph G and a number k, does G have an independent set of size k. This problem is NP-complete (NP-hard but in NP).

Now, in the given problem β corresponds to the Independent set decision problem. But there is a difference there. We have 5 instead of k. And this drastically changes the problem statement. We can now give a polynomial time **deterministic** algorithm for β.

- Find all vertex sets of size 5. We get ${}^{|V|}C_5$ such vertex sets
- For each of them check if there is any adjacent vertices. This check can be done in constant time if we use an Adjacency matrix representation

Thus the whole time complexity reduces to ${}^{|V|}C_5$ which is $O\left(|V|^5\right)$ and hence polynomial. (${}^{|V|}C_k$ is not polynomial but ${}^{|V|}C_5$ is).

Problem α is asking for an independent set of size $|V| - 4$. This is equivalent to asking if G has a vertex cover* of size 4. Following a similar approach as done for β this problem also can be done in polynomial time.

So, both α and β are in P.

D choice.

Vertex cover of a graph G is the set of vertices such that each edge of the graph is incident on atleast one of those vertices.

Independent Set and Vertex cover Reduction: https://www.cs.cmu.edu/~ckingsf/bioinfo-lectures/npcomplete.pdf

References

18 votes -- Arjun Suresh (330k points)

5.10.2 P Np Npc Nph: GATE CSE 2006 | Question: 31 top https://gateoverflow.in/994

✔ The only difference between **SHAM** and **DHAM**, in **SHAM** |V| is divisible by 3 which can be check in constant amount of time.

S,o the hardness of the two problem will the same. Next, finding hamiltonian cycle comes under **NPC** problem and **NPC** problem is a subset of **NPH**, so both are **NPH.**

So, option (**A**).

14 votes -- Vicky Bajoria (4.1k points)

5.10.3 P Np Npc Nph: GATE CSE 2009 | Question: 14 top https://gateoverflow.in/1306

✔ A problem which is in **P**, is also in **NP**- so, **A** is false. If problem can be solved deterministically in Polynomial time, then also we can't comment anything about P=NP, we just put this problem in P. So, **B** also false. **C** is TRUE because that is the definition of **NP**-complete.

D is false because all **NP** problems are not only decidable but decidable in polynomial time using a non-deterministic Turing machine.

19 votes -- shreya ghosh (2.8k points)

5.10.4 P Np Npc Nph: GATE CSE 2012 | Question: 4 top https://gateoverflow.in/36

✔ Answer is (**B**) $NP - complete \cap P = \phi$

Since, $P \neq NP$, there is at least one problem in NP, which is harder than all P problems. Lets take the hardest such problem, say X. Since, $P \neq NP, X \notin P$.

Now, by definition, $NP - complete$ problems are the hardest problems in NP and so X problem is in $NP - complete$. And being in NP, X can be reduced to all problems in $NP - complete$, making any other $NP - complete$ problem as $hard$ as X. So, since $X \notin P$, none of the other $NP - complete$ problems also cannot be in P.

29 votes -- Arjun Suresh (330k points)

5.10.5 P Np Npc Nph: GATE CSE 2013 | Question: 18 top https://gateoverflow.in/1440

✔ Cycle detection is graph is in P as it can be done using a graph traversal $(O(V + E))$

Ref: http://www.geeksforgeeks.org/detect-cycle-undirected-graph/

If a problem is in P then it is also in NP as P is a subset of NP. So, both **1** and **2** are TRUE.

Statement **3** is also true as $NP - Complete$ requires a problem to be in NP and for any problem in NP, we have a non-deterministic polynomial time algorithm.

So, answer is **A)** - 1, 2 and 3 are TRUE.

References

30 votes

-- Arjun Suresh (330k points)

5.11 Pumping Lemma (2) top

5.11.1 Pumping Lemma: GATE CSE 2019 | Question: 15 top

https://gateoverflow.in/302833

For $\Sigma = \{a, b\}$, let us consider the regular language $L = \{x \mid x = a^{2+3k} \text{ or } x = b^{10+12k}, k \geq 0\}$. Which one of the following can be a pumping length (the constant guaranteed by the pumping lemma) for L ?

A. 3
B. 5
C. 9
D. 24

Read Go Soln

gate2019-cse theory-of-computation pumping-lemma

Answer

5.11.2 Pumping Lemma: GATE IT 2005 | Question: 40 top

https://gateoverflow.in/3787

A language L satisfies the Pumping Lemma for regular languages, and also the Pumping Lemma for context-free languages. Which of the following statements about L is TRUE?

A. L is necessarily a regular language.
B. L is necessarily a context-free language, but not necessarily a regular language.
C. L is necessarily a non-regular language.
D. None of the above

gate2005-it theory-of-computation pumping-lemma easy

Answer

Answers: Pumping Lemma

5.11.1 Pumping Lemma: GATE CSE 2019 | Question: 15 top

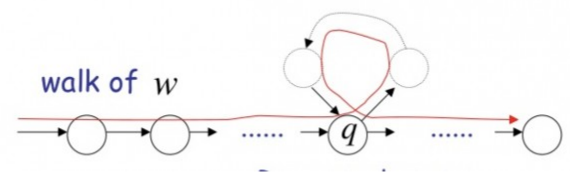
https://gateoverflow.in/302833

✔ Answer: D

Pumping length for a regular language makes sure that any string in that language with the length greater than pumping length has some repetition.

If we talk about DFA then it repeats some states. If we talk about regular grammar then it repeats atleast one nonterminal in derivation.

Let us suppose p is pumping length then any string w that has length greater than p must have repetition .



(In above DFA, for any string which has length $> p$ it has to go through the loop (repeat).)

$S \to aX \to \cdots \to aw'X \to \cdots \to w$

$S \to aX \to \cdots \to aw'X \to \cdots \to aw'w'X \to \cdots \to w$ (if repeat X 2 times)

If you notice we can generate infinitely many string by choosing number of repetitions to be 1,2,3 or any value.

Does this make any regular language an infinite language? No, we already know that regular languages are finite too.

What goes wrong here? Nothing.

For finite languages pumping length is greater than the longest string. And the definition of pumping length says that "Any string that is greater than p...". And there are no string which is greater than p which makes definition trivially true.

Now coming to the easiest part which is solving the question with clear definition in mind.

(First construct DFA)

Option A says that **any** string that has length 3 or greater has to have some repetition.

-Obviously false, b^{10} is a string in language and doesn't repeat any state.

Moreover any string with length greater than 10 needs to repeat states.

D is the correct answer.

52 votes -- Sachin Mittal (15.8k points)

Pumping lemma states a Deep Property that All Regular languages share. By Showing that a language does not have the property stated by Pumping lemma, we are guaranteed that It is Not Regular.

Pumping Lemma(PL) is a necessary condition for Regular languages but not sufficient condition

i.e. All Regular languages must satisfy this condition and some Non-regular languages also satisfy this condition.

So, **Regular Language → Satisfies Pumping Lemma**

$\therefore$ Contrapositive statement is **Doesn't Satisfy Pumping Lemma → Not Regular Languages**

Pumping lemma for Regular languages says

" If a language L is Regular,

then $\exists P \geq 1$, such that

$\forall$ $strings\ w \in L$, If $|w| \geq P$ then

$\exists x, y, z$, such that $w = xyz$

and $|xy| \leq P$

and $y \neq \in$ i.e. $|y| \geq 1$

and $\forall q \geq 0$, $xy^qz \in L$

In Words, If L is regular, then there's some magic number P(Called Pumping length). And if I take any string w that is at least as long as P, then I can break it up into three parts x, y,and z. Now, the length of xy is less than or equal to P, and also the length of y is greater than or equal to 1 and less than or equal to P.

In very simple words, If L is a regular language then there is some positive number P associated with L such that for all strings w of length greater than or equal to P, we can find some non-empty sub-string y in w within the first P symbols of w such that when we repeat y Zero or more times then the produced strings also belong to L.

Now, Coming to the language in the Question, this language consists of the following strings.

$L = \{a^2, a^5, a^8, a^{11}, a^{14}, \ldots\ldots\ldots, b^{10}, b^{22}, b^{34}, b^{46} \ldots\ldots\ldots\}$

Let me assume that Pumping length is $P = 10$ (Note that if Minimum pumping length for a language is x then any number $\geq x$ is also a Pumping length for the language)

Since, I have assumed that Pumping length is $P = 10$, so, For all strings whose length is ≥ 10 should be Pumped i.e. If w is a string of length ≥ 10 then within the first $P\ i.e.\ 10$ symbols of the string w, we should be able to find some non-empty substring y, such that we can repeat y Zero or More times and the resulted string still belongs to language L.

So, since $w = b^{10}$ has length ≥ 10 so, b^{10} must be pumped. But in b^{10} , If you take any non-empty substring and you remove that substring (Note that repeating substring y Zero times in w is equivalent to saying that remove y from the string w) then resulted string does not belong to the given language. So, Pumping length cannot be 10. And Since we know that if Minimum pumping length for a language is x then any number $\geq x$ is also a Pumping length for the language. So, Since 10 is Not a pumping length for L, so, any number ≤ 10 cannot be a Pumping length for L.

So, Option 1,2,3 can be eliminated.

Moreover, The minimum pumping length for this language is 12. So, any number ≥ 12 is a Pumping length for the given language.

Minimum Pumping length for the given language is P = 12 :

Take any string w of length ≥ 12.

Say, you take a^{14} then the non-empty substring y within the first $P\ i.e.12$ symbols can be taken as first three $a's$ i.e. aaa. When you repeat aaa Zero or more times, the resulted string still belongs to the language.

Similarly, for any $a^{2+3k}, k \geq 4$, y can be chosen as the first three $a's$ i.e. aaa. When you repeat aaa Zero or more times, the resulted string still belongs to the language.

Say, you take b^{22}, then the non-empty substring y within the first $P\ i.e.12$ symbols can be taken as first twelve $b's$ i.e. b^{12}. When you repeat b^{12} Zero or more times, the resulted string still belongs to the language.

Similarly, for any $b^{10+12k}, k \geq 1$, y can be chosen as the first twelve $b's$ i.e. b^{12}. When you repeat b^{12} Zero or more times, the resulted string still belongs to the language.

So, Minimum Pumping length for this given language L is **12. And So, any number ≥ 12 is a Pumping length for L.**

Extra Notes :

Coming to Minimum Pumping length(I'll use abbreviation MPL), MPL is the least possible value of P such that Pumping lemma is satisfied by the regular language.

Some Facts :

I recommend to go through the Proof of Pumping lemma for Regular languages or at least the Idea of Pumping lemma based on Pigeon hole principle. Here : http://www.ling.upenn.edu/courses/Fall_2003/ling106/PumpingLemma.pdf

1. If you see the intuitive Proof of Pumping lemma(based on pigeon hole principle), You will find that $MPL \leq n$ where n is the number of states in the minimal DFA accepting the regular language. So, If you draw mDFA(minimal DFA) for the given regular language and say the number of states in mDFA is n then you can say that MPL will be less than or equal to n.

2. MPL will always be strictly greater than the minimal string in the language. (Hint for Proof : Since y can be repeated Zero or more times, When you repeat y Zero times, the string length decreases by at least one symbol)

3. In the definition of Pumping lemma, $P \geq 1$ so, $MPL \geq 1$

4. If mDFA has a Dead state then $MPL \leq n-1$ (Hint for proof : If there is a Dead state, then all the strings accepted by the mDFA will be among the remaining $n-1$ states, so, if the language is Infinite, then the Loop must be between the remaining states.)

5. If language is Finite then MPL will be $x+1$ where x is the length of the longest string in the language.

6. If Minimum pumping length for a language is x then any number $\geq x$ is also a Pumping length for the language.

More examples to practice Minimum Pumping length : https://gateoverflow.in/303738/michael-sipser-exercise?show=303856#a303856

References

👍 95 votes -- Deepak Poonia (23.3k points)

5.11.2 Pumping Lemma: GATE IT 2005 | Question: 40 top

☛ https://gateoverflow.in/3787

✔ Answer is (**D**). If a language is regular, it definitely satisfies pumping lemma. But converse need not be true. If a language satisfies pumping lemma then it may or may not be regular.

👍 28 votes -- Rajat Sharma (475 points)

5.12 Pushdown Automata (13) top

5.12.1 Pushdown Automata: GATE CSE 1996 | Question: 13 top

☛ https://gateoverflow.in/2765

Let $Q = (\{q_1, q_2\}, \{a, b\}, \{a, b, \perp\}, \delta, \perp, \phi)$ be a pushdown automaton accepting by empty stack for the language which is the set of all nonempty even palindromes over the set $\{a, b\}$. Below is an incomplete specification of the transitions δ. Complete the specification. The top of the stack is assumed to be at the right end of the string representing stack contents.

1. $\delta(q_1, a, \perp) = \{(q_1, \perp a)\}$
2. $\delta(q_1, b, \perp) = \{(q_1, \perp b)\}$
3. $\delta(q_1, a, a) = \{(q_1, aa)\}$
4. $\delta(q_1, b, a) = \{(q_1, ab)\}$
5. $\delta(q_1, a, b) = \{(q_1, ba)\}$
6. $\delta(q_1, b, b) = \{(q_1, bb)\}$
7. $\delta(q_1, a, a) = \{(\dots, \dots)\}$
8. $\delta(q_1, b, b) = \{(\dots, \dots)\}$
9. $\delta(q_2, a, a) = \{(q_2, \epsilon)\}$
10. $\delta(q_2, b, b) = \{(q_2, \epsilon)\}$
11. $\delta(q_2, \epsilon, \perp) = \{(q_2, \epsilon)\}$

gate1996 theory-of-computation pushdown-automata normal descriptive

Answer

5.12.2 Pushdown Automata: GATE CSE 1997 | Question: 6.6 top

https://gateoverflow.in/2262

Which of the following languages over $\{a, b, c\}$ is accepted by a deterministic pushdown automata?

A. $\{wcw^R \mid w \in \{a, b\}^*\}$
B. $\{ww^R \mid w \in \{a, b, c\}^*\}$
C. $\{a^n b^n c^n \mid n \geq 0\}$
D. $\{w \mid w$ is a palindrome over $\{a, b, c\}\}$

Note: w^R is the string obtained by reversing $'w'$.

gate1997 theory-of-computation pushdown-automata easy

Answer

5.12.3 Pushdown Automata: GATE CSE 1998 | Question: 13 top

https://gateoverflow.in/1727

Let $M = (\{q_0, q_1\}, \{0, 1\}, \{z_0, X\}, \delta, q_0, z_0, \phi)$ be a Pushdown automation where δ is given by

$\delta(q_0, 1, z_0) = \{(q_0, Xz_0)\}$
$\delta(q_0, \epsilon, z_0) = \{(q_0, \epsilon)\}$
$\delta(q_0, 1, X) = \{(q_0, XX)\}$
$\delta(q_1, 1, X) = \{(q_1, \epsilon)\}$
$\delta(q_0, 0, X) = \{(q_1, X)\}$
$\delta(q_0, 0, z_0) = \{(q_0, z_0)\}$

a. What is the language accepted by this PDA by empty stack?
b. Describe informally the working of the PDA

gate1998 theory-of-computation pushdown-automata descriptive

Answer

5.12.4 Pushdown Automata: GATE CSE 1999 | Question: 1.6 top

https://gateoverflow.in/377

Let L_1 be the set of all languages accepted by a PDA by final state and L_2 the set of all languages accepted by empty stack. Which of the following is true?

A. $L_1 = L_2$
B. $L_1 \supset L_2$
C. $L_1 \subset L_2$
D. None

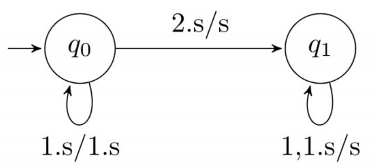
normal theory-of-computation gate1999 pushdown-automata

Answer

5.12.5 Pushdown Automata: GATE CSE 2000 | Question: 8 top

https://gateoverflow.in/679

A push down automation (pda) is given in the following extended notation of finite state diagram:



The nodes denote the states while the edges denote the moves of the pda. The edge labels are of the form d, s/s' where d is the input symbol read and s, s' are the stack contents before and after the move. For example the edge labeled $1, s/1.s$ denotes the move from state q_0 to q_0 in which the input symbol 1 is read and pushed to the stack.

A. Introduce two edges with appropriate labels in the above diagram so that the resulting pda accepts the language $\{x2x^R \mid x \in \{0,1\}^*, x^R \text{ denotes reverse of x}\}$, by empty stack.
B. Describe a non-deterministic pda with three states in the above notation that accept the language $\{0^n1^m \mid n \leq m \leq 2n\}$ by empty stack

gate2000-cse theory-of-computation descriptive pushdown-automata

Answer

5.12.6 Pushdown Automata: GATE CSE 2001 | Question: 6 top

☛ https://gateoverflow.in/747

Give a deterministic PDA for the language $L = \{a^n cb^{2n} \mid n \geq 1\}$ over the alphabet $\Sigma = \{a, b, c\}$. Specify the acceptance state.

gate2001-cse theory-of-computation normal pushdown-automata descriptive

Answer

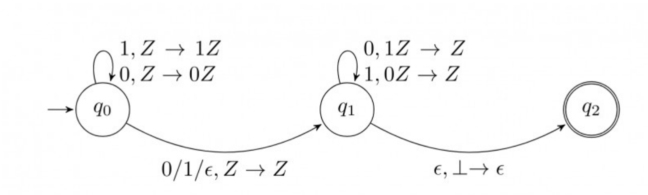
5.12.7 Pushdown Automata: GATE CSE 2015 Set 1 | Question: 51 top

☛ https://gateoverflow.in/8357

Consider the NPDA

$$\langle Q = \{q_0, q_1, q_2\}, \Sigma = \{0,1\}, \Gamma = \{0,1,\perp\}, \delta, q_0, \perp, F = \{q_2\}\rangle$$

, where (as per usual convention) Q is the set of states, Σ is the input alphabet, Γ is the stack alphabet, δ is the state transition function q_0 is the initial state, $\perp$ is the initial stack symbol, and F is the set of accepting states. The state transition is as follows:



HARD

Which one of the following sequences must follow the string 101100 so that the overall string is accepted by the automaton?

A. 10110
B. 10010
C. 01010
D. 01001

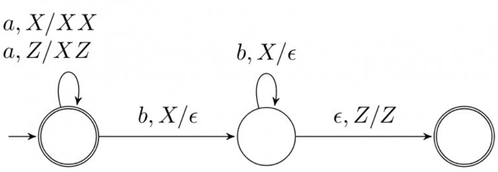
gate2015-cse-set1 theory-of-computation pushdown-automata normal

Answer

5.12.8 Pushdown Automata: GATE CSE 2016 Set 1 | Question: 43 top

☛ https://gateoverflow.in/39732

Consider the transition diagram of a PDA given below with input alphabet $\Sigma = \{a, b\}$ and stack alphabet $\Gamma = \{X, Z\}$. Z is the initial stack symbol. Let L denote the language accepted by the PDA



Which one of the following is **TRUE**?

A. $L = \{a^nb^n \mid n \geq 0\}$ and is not accepted by any finite automata
B. $L = \{a^n \mid n \geq 0\} \cup \{a^nb^n \mid n \geq 0\}$ and is not accepted by any deterministic PDA
C. L is not accepted by any Turing machine that halts on every input
D. $L = \{a^n \mid n \geq 0\} \cup \{a^nb^n \mid n \geq 0\}$ and is deterministic context-free

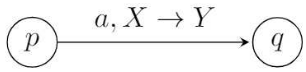
gate2016-cse-set1 theory-of-computation pushdown-automata normal

Answer

5.12.9 Pushdown Automata: GATE CSE 2021 Set 1 | Question: 51 top

https://gateoverflow.in/357400

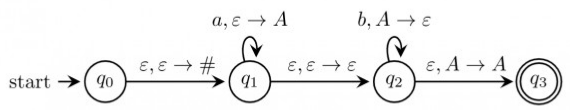
In a pushdown automaton $P = (Q, \Sigma, \Gamma, \delta, q_0, F)$, a transition of the form,



where $p, q \in Q$, $a \in \Sigma \cup \{\epsilon\}$, and $X, Y \in \Gamma \cup \{\epsilon\}$, represents

$$(q, Y) \in \delta(p, a, X).$$

Consider the following pushdown automaton over the input alphabet $\Sigma = \{a, b\}$ and stack alphabet $\Gamma = \{\#, A\}$.



The number of strings of length 100 accepted by the above pushdown automaton is ___________

gate2021-cse-set1 theory-of-computation pushdown-automata numerical-answers

Answer

5.12.10 Pushdown Automata: GATE IT 2004 | Question: 40 top

https://gateoverflow.in/3683

Let $M = (K, \Sigma, \Gamma, \Delta, s, F)$ be a pushdown automaton, where

$K = (s, f), F = \{f\}, \Sigma = \{a, b\}, \Gamma = \{a\}$ and
$\Delta = \{((s, a, \epsilon), (s, a)), ((s, b, \epsilon), (s, a)), ((s, a, a), (f, \epsilon)), ((f, a, a), (f, \epsilon)), ((f, b, a), (f, \epsilon))\}$.

Which one of the following strings is not a member of $L(M)$?

A. aaa
B. $aabab$
C. $baaba$
D. bab

gate2004-it theory-of-computation pushdown-automata normal

Answer

5.12.11 Pushdown Automata: GATE IT 2005 | Question: 38 top

https://gateoverflow.in/3785

Let P be a non-deterministic push-down automaton (NPDA) with exactly one state, q, and exactly one symbol, Z, in its stack alphabet. State q is both the starting as well as the accepting state of the PDA. The stack is initialized with one Z before the start of the operation of the PDA. Let the input alphabet of the PDA be Σ. Let $L(P)$ be the language accepted by the PDA by reading a string and reaching its accepting state. Let $N(P)$ be the language accepted by the PDA by reading a string and emptying its stack.
Which of the following statements is TRUE?

A. $L(P)$ is necessarily Σ^* but $N(P)$ is not necessarily Σ^*.
B. $N(P)$ is necessarily Σ^* but $L(P)$ is not necessarily Σ^*.
C. Both $L(P)$ and $N(P)$ are necessarily Σ^*.
D. Neither $L(P)$ nor $N(P)$ are necessarily Σ^*

gate2005-it theory-of-computation pushdown-automata normal

Answer

5.12.12 Pushdown Automata: GATE IT 2006 | Question: 31 top

https://gateoverflow.in/3570

Which of the following languages is accepted by a non-deterministic pushdown automaton (PDA) but NOT by a deterministic PDA?

A. $\{a^n b^n c^n \mid n \geq 0\}$

B. $\{a^l b^m c^n \mid l \neq m \text{ or } m \neq n\}$
C. $\{a^n b^n \mid n \geq 0\}$
D. $\{a^m b^n \mid m, n \geq 0\}$

gate2006-it theory-of-computation pushdown-automata normal

Answer

5.12.13 Pushdown Automata: GATE IT 2006 | Question: 33 top

https://gateoverflow.in/3572

Consider the pushdown automaton (PDA) below which runs over the input alphabet (a, b, c). It has the stack alphabet $\{Z_0, X\}$ where Z_0 is the bottom-of-stack marker. The set of states of the PDA is $(s, t, u, f\}$ where s is the start state and f is the final state. The PDA accepts by final state. The transitions of the PDA given below are depicted in a standard manner. For example, the transition $(s, b, X) \rightarrow (t, XZ_0)$ means that if the PDA is in state s and the symbol on the top of the stack is X, then it can read b from the input and move to state t after popping the top of stack and pushing the symbols Z_0 and X (in that order) on the stack.

$(s, a, Z_0) \rightarrow (s, XXZ_0)$
$(s, \epsilon, Z_0) \rightarrow (f, \epsilon)$
$(s, a, X) \rightarrow (s, XXX)$
$(s, b, X) \rightarrow (t, \epsilon)$
$(t, b, X) \rightarrow (t, \epsilon)$
$(t, c, X) \rightarrow (u, \epsilon)$
$(u, c, X) \rightarrow (u, \epsilon)$
$(u, \epsilon, Z_0) \rightarrow (f, \epsilon)$

The language accepted by the PDA is

A. $\{a^l b^m c^n \mid l = m = n\}$
B. $\{a^l b^m c^n \mid l = m\}$
C. $\{a^l b^m c^n \mid 2l = m + n\}$
D. $\{a^l b^m c^n \mid m = n\}$

gate2006-it theory-of-computation pushdown-automata normal

Answer

Answers: Pushdown Automata

5.12.1 Pushdown Automata: GATE CSE 1996 | Question: 13 top

https://gateoverflow.in/2765

✔ $\delta(q_1, a, b) = \{(q_2, ba)\}$ means from state q_1 on input a with stack top being b, the PDA moves to state q_2 and pushes a on top of stack.

So, here the missing transitions are at the middle of the input string:

$\delta(q_1, a, a) = \{(q_2, \epsilon)\}$
$\delta(q_1, b, b) = \{(q_2, \epsilon)\}$

Once middle is reached, now we should start popping. And so, we must go to state q_2 as well as pop the previous character on the stack. (The character before and after the middle must be same as the string is even length palindrome)

(This is a non-deterministic PDA)

23 votes -- Arjun Suresh (330k points)

5.12.2 Pushdown Automata: GATE CSE 1997 | Question: 6.6 top

https://gateoverflow.in/2262

✔

A. $\{wcw^R \mid w \in \{a, b\}^*\}$ // Can be realized using DPDA cz we know the center of the string that is c here //Hence Option **A** is ANS

B. $\{ww^R \mid w \in \{a, b, c\}^*\}$ // (set of even palindrom)NPDA bcz we can't find deterministically the center of palindrom string
cfg will $\rightarrow S \rightarrow aSa \mid bSb \mid cSc \mid \epsilon$

C. $\{a^nb^nc^n \mid n \geq 0\}$ // CSL

D. $\{w \mid w \text{ is a palindrome over } \{a,b,c\}\}$ // it is a cfl similar as option B
cfg will $\rightarrow S \rightarrow aSa \mid bSb \mid cSc \mid a \mid b \mid c \mid \epsilon$

27 votes -- Rajesh Pradhan (18.9k points)

5.12.3 Pushdown Automata: GATE CSE 1998 | Question: 13 top https://gateoverflow.in/1727

✔ q_0 is start state

$\delta(\mathbf{q_0, 0, Z_0}) = (\mathbf{q_0, Z_0})$

[Do Nothing operation, just read any no of 0's but do not keep in stack (any no of 0's because on reading 0's it remains on same state q_0)]

$\delta(\mathbf{q_0, 1, Z_0}) = (\mathbf{q_0, XZ_0})$ [Read first 1 and keep one X in stack]

$\delta(\mathbf{q_0, 1, X}) = (\mathbf{q_0, XX})$ [Read any no of 1's and keep one X for each 1 in stack]

$\delta(\mathbf{q_0, 0, X}) = (\mathbf{q_1, X})$

[Read single 0 and do nothing in stack, state changed from q_0 to q_1]

$\delta(\mathbf{q_1, 1, X}) = (\mathbf{q_1}, \epsilon)$

[Pop out one X from stack on reading each 1 on state q_1 (matching each 1 with the 1 read before single 0)]

$\delta(\mathbf{q_0}, \epsilon, \mathbf{Z_0}) = (\mathbf{q_0}, \epsilon)$

[stack is empty , inputs are accepted here ,that is , ϵ or any of 0's (we read earlier with Do Nothing operation)]

$\mathbf{L = \{0^m, m \geq 0\}}$

No input accept after reaching on q_1 because stack will remain with Z_0, stack initial symbol

Note : if we add one more transition $\delta(\mathbf{q_1}, \epsilon, \mathbf{Z_0}) = (\mathbf{q_1}, \epsilon)$, then L will be $\{0^m \cup 0^i1^j01^j, m, i, j \geq 0\}$

47 votes -- Praveen Saini (41.9k points)

5.12.4 Pushdown Automata: GATE CSE 1999 | Question: 1.6 top https://gateoverflow.in/377

✔ Answer to the question is (**A**) $L_1 = L_2$.
Reason is for any PDA which accepts by final state there is an equivalent PDA (equivalent means that accepts the same language) which accepts by empty stack and vice-verse.

Now, this is not the case for DPDAs.

> The set of languages accepted by a DPDA by empty stack is a strict subset of the set of languages accepted by a DPDA by final state.

It can also be said that set of languages accepted by a DPDA by empty stack is the set of languages accepted by a DPDA by final state and which has the prefix property.

A language has prefix property means if $w \in L$, then no proper prefix of $w \in L$.

From the above definition of prefix property it must be clear why DPDA by empty stack has this property. If any prefix of a word w (w in L) is in L means the stack should have been empty even before completely processing w. But, being a deterministic PDA, once the stack becomes empty, the DPDA accepts and halts. So, in no way can a DPDA accepts w and its prefix.

PS: A DPDA with acceptance by empty stack cannot even accept all regular languages- example a^*.

Good read: http://www.cs.ucr.edu/~jiang/cs150/slides4week7_PDA+EquivToCFG.pdf

References

94 votes -- gatecse (62.6k points)

5.12.5 Pushdown Automata: GATE CSE 2000 | Question: 8 top

☛ https://gateoverflow.in/679

(a) $x2x^R$

Say for some word 0112110 we have to push every thing into the stack till 2 . then we get 1 then 1 will be at top of stack so pop it or if get 0 then 0 will at top of stack so pop it. For any word of language it is applicable. 2 is a mark that tell now we have to pop 0 for 0 and 1 for 1.

So, on the edge q_0 to q_0 add $0, s/0.s$

and on edge q_1 to q_1 add $0, 0.s/s$

19 votes -- Praveen Saini (41.9k points)

Part(b)
(ϵ is used to denote pop operation, Z is the starting symbol on stack)

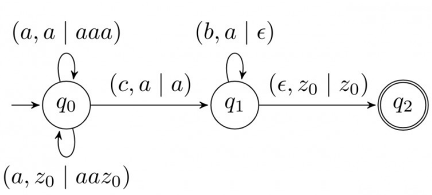
- $(q_0, 0, Z) \vdash (q_0, 0Z)$
- $(q_0, 0, Z) \vdash (q_0, 00Z)$
- $(q_0, 0, 0) \vdash (q_0, 000)$
- $(q_0, 0, 0) \vdash (q_0, 00)$
- $(q_0, 1, 0) \vdash (q_1, \epsilon)$
- $(q_1, 1, 0) \vdash (q_1, \epsilon)$
- $(q_0, \epsilon, Z) \vdash (q_0, \epsilon)$
- $(q_1, \epsilon, Z) \vdash (q_1, \epsilon)$

16 votes -- Saurav Shrivastava (1.3k points)

5.12.6 Pushdown Automata: GATE CSE 2001 | Question: 6 top

☛ https://gateoverflow.in/747

✔



$$L = \{a^n cb^{2n}\}$$
$$= \{acbb, aacbbbb, aaacbbbbbb, \ldots\}$$

Here, **c** acts as center. Push 2 a's for each **a** and after **c** start popping an **a** for each **b**. If stack is empty and string is finished we move to q_2 which is the acceptance state.

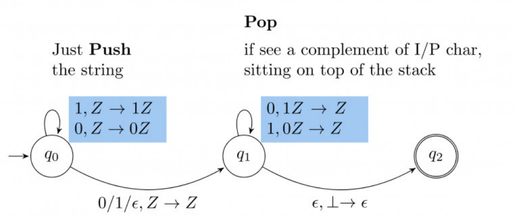
21 votes -- Saraswati Walijkar (291 points)

5.12.7 Pushdown Automata: GATE CSE 2015 Set 1 | Question: 51 top

☛ https://gateoverflow.in/8357

✔ Here, Z is used to represent the entire stack content except the top
Z is the string in Stack read from top to bottom. $1, Z \to 1Z$ means, on input symbol 1, the stack content changes from Z to $1Z$

In q_0 state, for '1', a '1' is pushed and for a '0' a '0' is pushed. In q_1 state, for a '0' a '1' is popped and for a '1' a '0' is popped. So, the given PDA is accepting all strings of of the form $x0x'_r$ or $x1x'_r$ or xx'_r, where x'_r is the reverse of the 1's complement of x. i.e.:



The given string 101100 has 6 letters and we are given 5 letter strings. So, $x0$ is done, with $x = 10110$. So,

$x_r' = (01001)_r = 10010.$

Answer is option **B.**

116 votes -- Arjun Suresh (330k points)

5.12.8 Pushdown Automata: GATE CSE 2016 Set 1 | Question: 43 top

https://gateoverflow.in/39732

✔ Strings accepted at I^{st} final state are a^n , $n \geq 0$ and strings accepted at II^{nd} final state are $a^n b^n$, $n \geq 0$ (actually $n \geq 1$ at this state, $n = 0$ already included at first state).

$L = \{a^n \mid n \geq 0\} \cup \{a^n b^n \mid n \geq 0\}$

Language is deterministic context-free accepted by DPDA (dpda is already given) and so by TM too, and not regular (as we need stack to implement it), and so cannot accepted by FA

Correct Answer: D

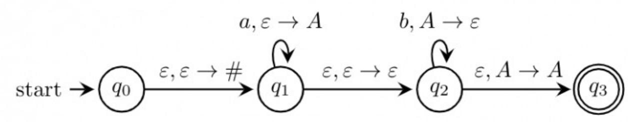
71 votes -- Praveen Saini (41.9k points)

5.12.9 Pushdown Automata: GATE CSE 2021 Set 1 | Question: 51 top

https://gateoverflow.in/357400

✔ **Correct Answer: 50**

A detailed analysis of the given Automaton(skip if you're comfortable with PDAs):



1. **PDAs** accept strings either by **emptying the stack** or by **reaching the final state** after reading the **entire input tape**. In this example, the transitions from initial state $q_0 \to q_1$ is marked by pushing $\#$ into the stack and is the only transition that involves $\#$ and can't be popped out again. So, there is no chance of acceptance by emptying the stack rather it's only by reaching the **final state**.
2. $a's$ can only be accepted in state q_1 and $b's$ only by state q_2 this implies the string must be of the form $\{a^m b^n\}$ for now.
3. To read $b's$ from the input tape, we've to pop-out A each time which's obtainable from accepting a's. So, the number $b's < a's$ for all the $b's$ to be accepted in this state.
4. To reach the final state we need at least one A to make the transition to the final state although it's funny about A not being spent.

Note: Reading ϵ from stack doesn't use up any of the stack contents just like ϵ transitions in a conventional FAs.

The given automaton accepts strings of the form $\{a^m b^n, \text{where } m \geq n+1\}$. And, it's given the question that $m + n = 100$.

So the possible set values in the form of (m, n) are $\{(100, 0), (99, 1) \ldots (51, 49)\}$ which in total there are 50. Try for $(m = 50, n = 50)$ to better understand why it isn't accepted.

2 votes -- Cringe is my middle name... (817 points)

5.12.10 Pushdown Automata: GATE IT 2004 | Question: 40 top

https://gateoverflow.in/3683

✔ Answer is D.

First of all the transition $(s, a, \epsilon), (s, a)$ means on input a, the PDA stays in same state and pushes a on stack irrespective of the current stack top value (ϵ means read nothing from stack and is not the stack bottom symbol). Like $epsilon$ moves in NFA, this makes the given PDA non-deterministic.

The language is like:

In start state a's or b's come, just push a's on stack except for the last a which is used to shift from "start state" to "final state" (non-determinism) without consuming any stack symbol. Now in "final state", for equal no's of a's and b's just pop a's from stack. This is the interpretation of transitions given for the language. So, it accepts strings of the form wab^n where $w \in (a + b)^*$ and $n < |w|$. This is violated only for option D and hence it is not in L.

For any string push 'a' on stack except for the final 'a' which will cause a move to final state and a pop. Then for every 'b', stack is popped. If stack becomes empty before string end, we reach a dead state and string is rejected. Else, accepted.

A. aaa: push a....push a....pop a Final State, ACCEPTED

B. $aabab$: push a....push a...push a...pop a...pop a Final State, ACCEPTED
C. $baaba$: push a...push a....push a....push a....pop a Final State, ACCEPTED
D. bab : push a....pop a....dead state REJECTED

PS: This PDA is an NPDA and acceptance is by FINAL State. If no valid move is given for any state in NPDA, then the corresponding transition goes to a dead state.

45 votes -- Rohan Mundhey (2.5k points)

5.12.11 Pushdown Automata: GATE IT 2005 | Question: 38 top

https://gateoverflow.in/3785

✔ **Answer is (D).**

In **NPDA** we may have a dead configuration. This mean we may not give any transition to any alphabet from this state.

We say that a string is accepted if PDA is in final state after reading the final symbol in the string or after it has read **'$'** symbol denoting end of the string and **it is in final state.**

Now coming to options:

Question never says that we have transitions defined for all the alphabet symbols in the PDA. Although it is ALREADY in the FINAL state we may not have ANY transition for any input symbol. In this case string will be rejected as it will never finish reading the string.

To sum up: A string is rejected in following two ways:

1. If no transition is defined for any configuration(this includes the final state as well because to accept the string we need the transition $(f, \$, \_) \rightarrow (f,_)$ in final state or accepting state where blank $('\_')$ denotes arbitrary stack symbol that does not matter because we are not accepting by EMPTY stack
2. If string enters a configuration for which no transition is defined, STRING is rejected.

So, option (D) is correct. Because the same way it may not empty the stack when it finishes reading the string.

50 votes -- Sandeep_Uniyal (6.5k points)

5.12.12 Pushdown Automata: GATE IT 2006 | Question: 31 top

https://gateoverflow.in/3570

✔ Ooption **B** is correct.

$L = \{a^l b^m c^n \mid l \neq m \text{ or } m \neq n\}$

$(q_0, a, Z_0) \rightarrow (q_0, aZ_0)$

$(q_0, a, a) \rightarrow (q_0, aa)$

$(q_0, b, a) \rightarrow (q_1, \epsilon), (q_2, ba)$

[here it is NPDA where we have to check $l \neq m$ or $m \neq n$; for $l \neq m$ we need to pop a for b; for $m \neq n$ we need to keep b in stack so that we can pop b for c]

$(q_1, b, a) \rightarrow (q_1, \epsilon)$

$(q_1, c, a) \rightarrow (q_f, \epsilon)$

$(q_1, b, Z_0) \rightarrow (q_f, \epsilon)$

$(q_2, b, b) \rightarrow q_2, bb)$

$(q_2, c, b) \rightarrow (q_3, \epsilon)$

$(q_3, c, b) \rightarrow (q_3, \epsilon)$

$(q_3, c, a) \rightarrow (q_f, \epsilon)$

$(q3, \epsilon, b) \rightarrow (q_f, \epsilon)$

(A) is wrong as it is not context free

(D) a^*b^* is regular, so must have **DFA** , and so an equivalent DPDA

(C) can be accepted using DPDA

29 votes -- Praveen Saini (41.9k points)

5.12.13 Pushdown Automata: GATE IT 2006 | Question: 33 top

https://gateoverflow.in/3572

✔ For every a we put two X in stack [at state s]

After that for every b we pop out one X [reach to state t (getting b after a)]

After that for every c we pop out one X [reach to state u (getting c after b)]

If all X are popped out then reached to final state f, mean for every b there is a, for every c there is a.

a was followed by b and b was followed by c [state s to t, t to u, u to f, final]

means sum of no of b's and no of c's $=$ twice of no of a's [one a for one b, one a for one c]

i.e. $\{a^l b^m c^n \mid 2l = m + n\}$

Correct Answer: C

35 votes -- Praveen Saini (41.9k points)

5.13 Recursive And Recursively Enumerable Languages (13) top

5.13.1 Recursive And Recursively Enumerable Languages: GATE CSE 1990 | Question: 3-vi top

Recursive languages are:

A. A proper superset of context free languages.
B. Always recognizable by pushdown automata.
C. Also called type 0 languages.
D. Recognizable by Turing machines.

gate1990 normal theory-of-computation turing-machine recursive-and-recursively-enumerable-languages multiple-selects

Answer

5.13.2 Recursive And Recursively Enumerable Languages: GATE CSE 2003 | Question: 13 top

Nobody knows yet if $P = NP$. Consider the language L defined as follows.

$$L = \begin{cases} (0+1)^* & \text{if } P = NP \\ \phi & otherwise \end{cases}$$

Which of the following statements is true?

A. L is recursive
B. L is recursively enumerable but not recursive
C. L is not recursively enumerable
D. Whether L is recursively enumerable or not will be known after we find out if $P = NP$

gate2003-cse theory-of-computation normal recursive-and-recursively-enumerable-languages

Answer

5.13.3 Recursive And Recursively Enumerable Languages: GATE CSE 2003 | Question: 15 top

If the strings of a language L can be effectively enumerated in lexicographic (i.e., alphabetic) order, which of the following statements is true?

A. L is necessarily finite
B. L is regular but not necessarily finite
C. L is context free but not necessarily regular
D. L is recursive but not necessarily context-free

theory-of-computation gate2003-cse normal recursive-and-recursively-enumerable-languages

Answer

5.13.4 Recursive And Recursively Enumerable Languages: GATE CSE 2005 | Question: 56 top

Let L_1 be a recursive language, and let L_2 be a recursively enumerable but not a recursive language. Which one of the following is TRUE?

A. L_1' is recursive and L_2' is recursively enumerable
B. L_1' is recursive and L_2' is not recursively enumerable
C. L_1' and L_2' are recursively enumerable
D. L_1' is recursively enumerable and L_2' is recursive

gate2005-cse theory-of-computation recursive-and-recursively-enumerable-languages easy

Answer

5.13.5 Recursive And Recursively Enumerable Languages: GATE CSE 2008 | Question: 13, ISRO2016-36 top

If L and $\bar{L}$ are recursively enumerable then L is

A. regular
B. context-free
C. context-sensitive
D. recursive

gate2008-cse theory-of-computation easy isro2016 recursive-and-recursively-enumerable-languages

Answer

5.13.6 Recursive And Recursively Enumerable Languages: GATE CSE 2008 | Question: 48 top

Which of the following statements is false?

A. Every NFA can be converted to an equivalent DFA
B. Every non-deterministic Turing machine can be converted to an equivalent deterministic Turing machine
C. Every regular language is also a context-free language
D. Every subset of a recursively enumerable set is recursive

gate2008-cse theory-of-computation easy recursive-and-recursively-enumerable-languages

Answer

5.13.7 Recursive And Recursively Enumerable Languages: GATE CSE 2010 | Question: 17 top

Let L_1 be the recursive language. Let L_2 and L_3 be languages that are recursively enumerable but not recursive. Which of the following statements is not necessarily true?

A. $L_2 - L_1$ is recursively enumerable.
B. $L_1 - L_3$ is recursively enumerable.
C. $L_2 \cap L_3$ is recursively enumerable.
D. $L_2 \cup L_3$ is recursively enumerable.

gate2010-cse theory-of-computation recursive-and-recursively-enumerable-languages decidability normal

Answer

5.13.8 Recursive And Recursively Enumerable Languages: GATE CSE 2014 Set 1 | Question: 35 top

Let L be a language and $\bar{L}$ be its complement. Which one of the following is **NOT** a viable possibility?

A. Neither L nor $\bar{L}$ is recursively enumerable $(r.e.)$.
B. One of L and $\bar{L}$ is r.e. but not recursive; the other is not r.e.
C. Both L and $\bar{L}$ are r.e. but not recursive.
D. Both L and $\bar{L}$ are recursive.

gate2014-cse-set1 theory-of-computation easy recursive-and-recursively-enumerable-languages

Answer

5.13.9 Recursive And Recursively Enumerable Languages: GATE CSE 2014 Set 2 | Question: 16 top

Let $A \leq_m B$ denotes that language A is mapping reducible (also known as many-to-one reducible) to language B. Which one of the following is FALSE?

A. If $A \leq_m B$ and B is recursive then A is recursive.
B. If $A \leq_m B$ and A is undecidable then B is undecidable.
C. If $A \leq_m B$ and B is recursively enumerable then A is recursively enumerable.
D. If $A \leq_m B$ and B is not recursively enumerable then A is not recursively enumerable.

gate2014-cse-set2 theory-of-computation recursive-and-recursively-enumerable-languages normal

Answer

5.13.10 Recursive And Recursively Enumerable Languages: GATE CSE 2015 Set 1 | Question: 3 top

For any two languages L_1 and L_2 such that L_1 is context-free and L_2 is recursively enumerable but not recursive, which of the following is/are necessarily true?

I. $\bar{L}_1$ (Compliment of L_1) is recursive
II. $\bar{L}_2$ (Compliment of L_2) is recursive
III. $\bar{L}_1$ is context-free
IV. $\bar{L}_1 \cup L_2$ is recursively enumerable

A. I only
B. III only
C. III and IV only
D. I and IV only

gate2015-cse-set1 theory-of-computation recursive-and-recursively-enumerable-languages normal

Answer

5.13.11 Recursive And Recursively Enumerable Languages: GATE CSE 2016 Set 1 | Question: 44 top

Let X be a recursive language and Y be a recursively enumerable but not recursive language. Let W and Z be two languages such that $\overline{Y}$ reduces to W, and Z reduces to $\overline{X}$ (reduction means the standard many-one reduction). Which one of the following statements is TRUE?

A. W can be recursively enumerable and Z is recursive.
B. W can be recursive and Z is recursively enumerable.
C. W is not recursively enumerable and Z is recursive.
D. W is not recursively enumerable and Z is not recursive.

gate2016-cse-set1 theory-of-computation easy recursive-and-recursively-enumerable-languages

Answer

5.13.12 Recursive And Recursively Enumerable Languages: GATE CSE 2016 Set 2 | Question: 44 top

Consider the following languages.

- $L_1 = \{\langle M \rangle \mid M \text{ takes at least 2016 steps on some input}\}$,
- $L_2 = \{\langle M \rangle \mid M \text{ takes at least 2016 steps on all inputs}\}$ and
- $L_3 = \{\langle M \rangle \mid M \text{ accepts } \epsilon\}$,

where for each Turing machine M, $\langle M \rangle$ denotes a specific encoding of M. Which one of the following is TRUE?

A. L_1 is recursive and L_2, L_3 are not recursive
B. L_2 is recursive and L_1, L_3 are not recursive
C. L_1, L_2 are recursive and L_3 is not recursive
D. L_1, L_2, L_3 are recursive

gate2016-cse-set2 theory-of-computation recursive-and-recursively-enumerable-languages

Answer

5.13.13 Recursive And Recursively Enumerable Languages: GATE CSE 2021 Set 1 | Question: 12 top

Let $\langle M\rangle$ denote an encoding of an automaton M. Suppose that $\Sigma = \{0,1\}$. Which of the following languages is/are NOT recursive?

A. $L = \{\langle M\rangle \mid M \text{ is a DFA such that } L(M) = \emptyset\}$
B. $L = \{\langle M\rangle \mid M \text{ is a DFA such that } L(M) = \Sigma^*\}$
C. $L = \{\langle M\rangle \mid M \text{ is a PDA such that } L(M) = \emptyset\}$
D. $L = \{\langle M\rangle \mid M \text{ is a PDA such that } L(M) = \Sigma^*\}$

gate2021-cse-set1 multiple-selects theory-of-computation recursive-and-recursively-enumerable-languages

Answer

Answers: Recursive And Recursively Enumerable Languages

5.13.1 Recursive And Recursively Enumerable Languages: GATE CSE 1990 | Question: 3-vi top

✔
A. A proper superset of context free languages. **TRUE** Since there are languages which are not CFL still Recursive
B. Always recognizable by pushdown automata. **FALSE**
C. Also called type 0 languages. **FALSE** R.E languages are actually type-0 languages.
D. Recognizable by Turing machines **TRUE**

22 votes -- Prajwal Bhat (7.6k points)

5.13.2 Recursive And Recursively Enumerable Languages: GATE CSE 2003 | Question: 13 top

✔ Correct Option: **A**

L is recursive. If $P = NP$, L is Σ^* which is recursive (in fact regular). If not, $L = \phi$ which is again recursive. So, in both cases L is recursive.

40 votes -- Arjun Suresh (330k points)

5.13.3 Recursive And Recursively Enumerable Languages: GATE CSE 2003 | Question: 15 top

✔ Answer is (**D**) L is recursive but not necessarily Regular or not even context-free.

Since, the strings of L can be enumerated it means L is recursively enumerable. That is we have a TM which accepts all strings in L. Now, to be recursive the TM should reject all strings not in L. Since, the strings of the language can be enumerated in lexicographic order, it's easy to do this. For any word w, if we see a word in the enumeration which is lexicographically higher than w but no w, it means w is not in the language. This makes L '"recursive"'.

Now, why L need not be context free or regular? Consider

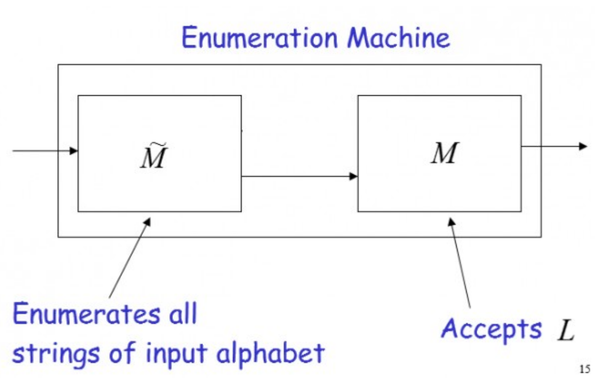
$$L = \{a^n b^n c^n \mid n \geq 0\}$$

The strings of this language can be enumerated in lexicographic order. But we know L is not context free as no PDA can accept L.

113 votes -- Arjun Suresh (330k points)

Very important line-
REC has lexicographic enumeration procedure but RE do not have lexicographic enumeration procedure (Of course, RE has enumeration procedure, that is why it is called Recursively enumerable language but it need not to be lexicographic)
(Throughout this answer, whenever I write RE it means "RE but Not REC")

Why so ?
-for REC we have Halting Turing machine.
Enumeration procedure for REC-
I can give each string in HTM and wait for some time, either HTM will accept it or reject it. If accept it then print string and move on to next string, if reject then move to next string without printing.
Of course i can give strings in lexicographic order and this procedure will enumerate in the same order in which i give input.



[15]

Here $\tilde{M}$ generates all strings in lexicographic order (or in whatever order we want) and M also outputs in same order.

Can we use same procedure for RE ? (again RE means "RE but not REC")
NO!, this procedure won't work for RE, Why?

That enumeration procedure looks like this-

Repeat: $\tilde{M}$ generates a string w
    M checks if $w \in L$
        Yes: Print w
        No: Ignore
This procedure won't work for RE.
Problem: If $w \notin L$ M may loop forever

Now, The question is, How to enumerate RE ?, Which is best procedure for enumerating RE ?
-We all know that there is a procedure to enumerate RE, but we never bother about how this procedure looks like :D

I have one idea, not sure if this works or not, Let's see-
-I give all strings to TM simultaneously, and as soon as one string is accepted by TM, It will print.
Say, 1st string is taking 500 steps and 2nd string is taking 5 steps then after 5 steps it will print 2nd string, and after 500 steps it will print 1st string.
Seems like it will work and will enumerate strings of RE language, but How can i give all strings simultaneously?
-If i modify the procedure mentioned above then it will pretend like all strings are given simultaneously.
$\tilde{M}$ generates first string w_1
    M executes first step on w_1
$\tilde{M}$ generates second string w_2
    M executes first step on w_2
        Second step on w_1
$\tilde{M}$ generates third string w_3
    M executes first step on w_3
    M executes second step on w_2
        Third step on w_1

And So on....

If for string machine halts in a final state then it prints on the output.

This procedure enumerates in random order.
Say, 1st string is taking 1000 steps of TM and 5th takes only 2 steps. Then it prints 5th string first.

Now if i ask you "A language L can be enumerated by length if and only if it is recursive.", Then You must agree that this statement is also true.
Be it any order, if u can enumerate in that order then that language can not be RE.

39 votes — -- Sachin Mittal (15.8k points)

5.13.4 Recursive And Recursively Enumerable Languages: GATE CSE 2005 | Question: 56 top

✔ L_1 being recursive, we have a TM M for L_1 which accepts all words in L_1 and rejects all words in L_1'. So, this TM also works for L_1' by changing the accept and reject states. Thus L_1' is recursive.

L_2 being recursively enumerable but not recursive means TM for L_2 can accept all words in L_2 but cannot reject all words not in $L_2 \Rightarrow$ TM for L_2' cannot exist (as otherwise TM for L_2 could simulate the moves of that TM to reject words in L_2') $\Rightarrow L_2$' is not recursively enumerable. So, (**B**).

34 votes -- Arjun Suresh (330k points)

5.13.5 Recursive And Recursively Enumerable Languages: GATE CSE 2008 | Question: 13, ISRO2016-36 top

✔ (**D**) recursive

L is recursively enumerable means a TM accepts all strings in L. $\bar{L}$ is recursively enumerable means a TM accepts all strings in $\bar{L}$. So, we can always decide if a string is in L or not, making L recursive.

http://goo.gl/RtV8MO

References

37 votes -- Keith Kr (4.5k points)

5.13.6 Recursive And Recursively Enumerable Languages: GATE CSE 2008 | Question: 48 top

✔ There exists a set of languages which is **RE** but not **REC** (i.e. Recursively Enumerable but not Recursive), this set is a subset of **RE** but is Not Recursive.

Option **D** tells us that every subset of **RE** is **REC** this is false.
Hence, **option D** is chosen.

25 votes -- Amar Vashishth (25.2k points)

5.13.7 Recursive And Recursively Enumerable Languages: GATE CSE 2010 | Question: 17 top

✔ Recursively enumerable languages are closed under union and intersection. So, lets consider each option

A. $L_2 - L_1 = L_2 \cap \overline{L_1}$
   Recursive languages are closed under complement, and so $\overline{L_1}$ is recursive and hence recursively enumerable also. So, $L_2 \cap \overline{L_1}$ is recursively enumerable is always TRUE.
B. $L_1 - L_3 = L_1 \cap \overline{L_3}$
   Recursively enumerable languages are not closed under complement. So, $\overline{L_3}$ may or may not be recursively enumerable and hence we can't say anything if $L_1 \cap \overline{L_3}$ is recursively enumerable or not.
C. Intersection of two recursively enumerable languages is always recursively enumerable(RE closed under intersection).
D. Union of two recursively enumerable languages is always recursively enumerable(RE closed under union).

For verifying closure properties:
http://gatecse.in/wiki/Closure_Property_of_Language_Families

Correct Answer: B

References

77 votes -- Arjun Suresh (330k points)

5.13.8 Recursive And Recursively Enumerable Languages: GATE CSE 2014 Set 1 | Question: 35 top

✔ (**C**) is not possible. If L is re we have a TM that accepts string in L. If L' is re, we have a TM that accepts strings in L'. So, using both these TMs we can make a new TM M which accepts strings in L and rejects strings in L' - that is M decides L, making L recursive.

34 votes -- Arjun Suresh (330k points)

5.13.9 Recursive And Recursively Enumerable Languages: GATE CSE 2014 Set 2 | Question: 16 top

✔ $A \leq_m B$ means A cannot be harder than B. (Since A can be reduced to B, instead of deciding A, we can now decide B)

So, the first 3 options are correct. Option (**D**) is false, as B is not recursively enumerable doesn't guarantee A is not recursively enumerable.

43 votes -- Arjun Suresh (330k points)

5.13.10 Recursive And Recursively Enumerable Languages: GATE CSE 2015 Set 1 | Question: 3 top

✔ **Answer is D**.

L_1 is context-free and hence recursive also. Recursive set being closed under complement, L_1' will be recursive.

L_1' being recursive it is also recursively enumerable and Recursively Enumerable set is closed under Union. So, $L_1' \cup L_2$ is recursively enumerable.

Context free languages are not closed under complement, so III is false

Recursive set is closed under complement. So, if L_2' is recursive, (L_2')' $= L_2$ is also recursive which is not the case here. So, II is also false.

36 votes -- Arjun Suresh (330k points)

5.13.11 Recursive And Recursively Enumerable Languages: GATE CSE 2016 Set 1 | Question: 44 top

✔ X is recursive language, so $\overline{X}$ is also recursive.

Y is recursively enumerable,but not recursive so $\overline{Y}$ is not recursively enumerable language.

$A \leq B$, (A is reducible to B) , i. e, solving A cannot be "harder" than solving B.

1. If A is reducible to B, and B is decidable, then A is decidable.
   i) if A is reducible to B, and B is recursive, then A is recursive.
2. If A is undecidable and reducible to B, then B is undecidable.
   i) if B is recursively enumerable, and A is reducible to B, then A is recursively enumerable.
   ii) if A is not recursively enumerable, and reducible to B, then B is not recursively enumerable.

Now Back to question.

$\overline{Y}$ is not recursively enumerable, and reducible to W, then W is not recursively enumerable (using 2(ii)).

Z is reducible to $\overline{X}$ and $\overline{X}$ is recursive, then Z is recursive (using 1(i)).

Option **C** is correct.

81 votes -- Praveen Saini (41.9k points)

5.13.12 Recursive And Recursively Enumerable Languages: GATE CSE 2016 Set 2 | Question: 44 top

✔ L_3 is not recursive as it asks if $L(M)$ contains ϵ which is a non-trivial property of r.e. languages and hence undecidable as per Rice's theorem.

L_1 and L_2 are slightly trickier as these are not describing properties of recursively enumerable languages, but rather of Turing machines. So, we can see if there is some procedure for deciding these.

For L_1 we can give the TM an input of length 2016. Now, it should at least make 2016 steps or reach the halt state before completing the input processing. The second case is possible only if the TM reaches a halt state before reaching the end of string (blank) of input, for all possible inputs of length at least 2016 and can be decided. So, we can be sure that otherwise TM will have at least 2016 steps making L_1 recursive.

L_2 is recursive and it is more easier to prove. For the complement of L_2 we need M to make less than 2016 steps for some input and we can just give it all possible inputs of length less than 2016 and see if it reaches a halt state within 2016 steps. Thus complement of L_2 is recursive $\implies L_2$ is recursive.

So, answer here is **C**.

86 votes -- Arjun Suresh (330k points)

One more possible approach:

L_3 is not recursive (Can be proved using Rice theorem).

Lets talk about L_1 and L_2, and let me take L_2 first

$L_2 = \{\langle M \rangle \mid M$ takes at least 2016 steps on all inputs $\}$,

I want to check if for all strings in Σ^* M takes more than or equal to 2016 steps or not.

First of all I will restrict number of steps in M to 2016, and I will never run M more than 2016 steps. Because for any string, if M halts (accepts then halts or rejects then halts, does not matter) in less than 2016 steps then that string is not in L_2. And if M does not halt within 2016 steps (after 2016 steps, I am not interested whether M is in infinite loop or will halt eventually) then string is in L_2.

$\implies$ Number of steps to be run in M is not more than 2016.

Since, we bound the number of steps that M runs on an input, then there is no point on looking at any strings that are longer than that number, since if a TM is allowed to run for at most c steps, it is not possible for that TM to "process" any input symbol beyond the c^{th} symbol!

$\implies$ Length of input string is less than 2016. (If I can decide for these strings then L_2 is Recursive otherwise not Recursive) And there are finite strings having length less than 2016.

Now my task reduces to : "Take each string in this finite set and run M for finite number of steps"

The number of possible inputs is finite, and the number of steps M runs on each input is finite, therefore M is guaranteed to halt and decide the language. Hence L2 is recursive.

(If we can decide for all inputs then we can also decide for some inputs therefore L_1 is also recursive (Reduction), but we can think of L_1 as an independent problem)

$L_1 = \{\langle M \rangle \mid M$ takes at least 2016 steps on some input$\}$,

I want to check if there exist any string in Σ^* for which M takes more than or equal to 2016 steps.

With same reasoning I can say that we will run M for finite number of steps and input string set is also finite. The only difference is, we can stop giving input once we find any string taking at least 2016 steps, whereas in L_2 we have to check for all set of input strings length less than 2016.

Therefore, L_1 is also recursive.

L_1, L_2 : Recursive

L_3 : Not Recursive.

C is answer.

Ref: Problem number one in this pdf: https://www.cs.rice.edu/~nakhleh/COMP481/final_review_sp06_sol.pdf .

References

121 votes -- Sachin Mittal (15.8k points)

5.13.13 Recursive And Recursively Enumerable Languages: GATE CSE 2021 Set 1 | Question: 12 top

✔ **Correct Option: D**

A & B are **recursive**, since for every **regular language**, there exists a **unique minima**l **DFA** and we've a **minimization procedure** for the same. We could therefore compare any two **regular languages** which makes options A and B **recursive** (corresponding problem is decidable)

Option C is **recursive**.
For every **PDA** there is a corresponding **CFG** and vice versa. Moreover they're inter-convertible (see the references). So, we can convert the given **PDA** to its equivalent **CFG**. Then, we have algorithms to remove **empty, unit and useless productions.** If the language of the given **PDA** is **empty** then the **Start Symbol** would be **useless** (not generating any strings) which is **decidable** using an algorithm.

Option D is the Universality problem of CFLs and it is not **decidable** (not even **semi-decidable**). So, the given language is neither **recursive nor recursively enumerable.**

References:

1. https://my.eng.utah.edu/~cs3100/lectures/l18/pda-notes.pdf (pg 257)
2. https://www.cse.cuhk.edu.hk/~siuon/csci3130-f17/slides/lec11.pdf (slide 7)
3. https://github.com/typesAreSpaces/PDA-to-CFG (if you wanna go crazy)
4. https://gatecse.in/grammar-decidable-and-undecidable-problems/ (do check it out)
5. https://gatecse.in/closure-property-of-language-families/

Try coming up with **counter-examples** for undecidability, and try to at least remember for **RG**, **CFG** and **UG**.

Problem	RL	DCFL	CFL	CSL	RL	REL
Is $w \in L$?(Membership problem)	D	D	D	D	D	UD
Is $L = \phi$?(Emptiness problem)	D	D	D	UD	UD	UD
Is $L = \Sigma^*$?(Completeness problem)	D	D	UD	UD	UD	UD
Is $L_1 = L_2$?(Equality problem)	D	D	UD	UD	UD	UD
Is $L_1 \subseteq L_2$?(Subset problem)	D	UD	UD	UD	UD	UD
Is $L_1 \cap L_2 = \phi$?	D	UD	UD	UD	UD	UD
Is L finite? (finiteness problem)	D	D	D	UD	UD	UD
Is complement of 'L' a language of the same class?	D	D	UD	D	D	UD
Is intersection of two languages of the same class?	D	UD	UD	D	D	D
Is L a regular Language?	D	D	UD	UD	UD	UD

PS:

Some of the references I presented here notably 1 & 3 do go a stretch beyond what's necessary I hope someone will find them useful which is why I added them. I believe it's suffice to know that such an algorithm does exist but beyond that is unnecessary (arguable though).

References

4 votes

-- Cringe is my middle name... (817 points)

5.14 Recursive Recursively Enumerable Languages (1) top

5.14.1 Recursive Recursively Enumerable Languages: GATE CSE 2021 Set 1 | Question: 39 top

For a Turing machine M, $\langle M \rangle$ denotes an encoding of M. Consider the following two languages.

$$L_1 = \{\langle M \rangle \mid M \text{ takes more than 2021 steps on all inputs}\}$$
$$L_2 = \{\langle M \rangle \mid M \text{ takes more than 2021 steps on some input}\}$$

Which one of the following options is correct?

A. Both L_1 and L_2 are decidable
B. L_1 is decidable and L_2 is undecidable
C. L_1 is undecidable and L_2 is decidable
D. Both L_1 and L_2 are undecidable

gate2021-cse-set1 theory-of-computation recursive-recursively-enumerable-languages decidability easy

Answer

Answers: Recursive Recursively Enumerable Languages

5.14.1 Recursive Recursively Enumerable Languages: GATE CSE 2021 Set 1 | Question: 39 top

✔ $L_1 = \{\langle M \rangle \mid M \text{ takes more than 2021 steps on all inputs}\}$

$L_2 = \{\langle M \rangle \mid M \text{ takes more than 2021 steps on some input}\}$

Here, both L_1 and L_2 are decidable as we can have a systematic procedure in deciding them (correctly saying if an input is in L or not)

For both L_1 and L_2 we have to monitor the TM for $2021 + 1$ steps for all possible inputs of size 2021 (if the input set is having k alphabets we will have k^{2020} possible strings which is still a finite number.)

If for all the inputs M is taking more than 2021 steps, then it means for all larger strings also it must take more than 2021 steps and we can answer "yes" for L_1 or else "no".

If for none of the inputs M is taking more than 2021 steps then it means even for any larger string M won't be taking more than 2021 steps. So, we can answer "no" for L_2 or else "yes".

Thus we correctly decided both L_1 and L_2.

NOTE: There is intersection between

- "yes" case of L_1 and "yes" case of L_2
- "no" case of L_1 and "yes" case of L_2
- "no" case of L_2 and "no" case of L_1

but not between the "yes" case of L_1 and "no" case of L_2.

Correct Option: A.

1 votes -- Arjun Suresh (330k points)

Both of them have to be decidable. It is known that wd have a Turing machine that can simulate the moves of any Turing machine with an input applied to it.

In 2021 steps, note that the Tape head of a Deterministic (even if its non deterministic, we can do dovetailing) moves at the most 2021 steps to the right. Therefore for both L1 and L2 we can can do an exhaustive simulation of the TM coded as <M> over all the inputs of length upto 2022(anything greater than or equal to 2021 should do) and see if the predicate of L1 is satisfied for problem L1, and predicate of L2 is satisfied for problem L2

For L1:

M takes more than 2021 steps to halt on all inputs (iff) M takes more than 2021 steps on all the inputs of length less than or equal to 2022

For L2:

M takes more than 2021 steps to halt on some input (iff) M takes more than 2021 steps on some input of length less than or equal to 2022

2 votes -- aniketh317 (103 points)

5.15 Regular Expressions (24) top

5.15.1 Regular Expressions: GATE CSE 1987 | Question: 10d top https://gateoverflow.in/82455

Give a regular expression over the alphabet $\{0,1\}$ to denote the set of proper non-null substrings of the string 0110.

gate1987 theory-of-computation regular-expressions descriptive

Answer

5.15.2 Regular Expressions: GATE CSE 1991 | Question: 03,xiii top https://gateoverflow.in/527

Let $r = 1(1+0)^*, s = 11^*0$ and $t = 1^*0$ be three regular expressions. Which one of the following is true?

A. $L(s) \subseteq L(r)$ and $L(s) \subseteq L(t)$
B. $L(r) \subseteq L(s)$ and $L(s) \subseteq L(t)$
C. $L(s) \subseteq L(t)$ and $L(s) \subseteq L(r)$
D. $L(t) \subseteq L(s)$ and $L(s) \subseteq L(r)$
E. None of the above

gate1991 theory-of-computation regular-expressions normal multiple-selects

Answer

5.15.3 Regular Expressions: GATE CSE 1992 | Question: 02,xvii top https://gateoverflow.in/575

Which of the following regular expression identities is/are TRUE?

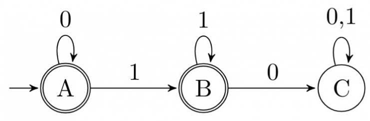
A. $r^{(*)} = r^*$
B. $(r^*s^*) = (r+s)^*$
C. $(r+s)^* = r^* + s^*$
D. $r^*s^* = r^* + s^*$

gate1992 theory-of-computation regular-expressions easy multiple-selects

Answer

5.15.4 Regular Expressions: GATE CSE 1994 | Question: 2.10 top https://gateoverflow.in/2477

The regular expression for the language recognized by the finite state automaton of figure is ________



gate1994 theory-of-computation finite-automata regular-expressions easy fill-in-the-blanks

Answer

5.15.5 Regular Expressions: GATE CSE 1995 | Question: 1.9 , ISRO2017-13 top https://gateoverflow.in/2596

In some programming language, an identifier is permitted to be a letter followed by any number of letters or digits. If L and D denote the sets of letters and digits respectively, which of the following expressions defines an identifier?

A. $(L+D)^+$
B. $(L.D)^*$
C. $L(L+D)^*$
D. $L(L.D)^*$

gate1995 theory-of-computation regular-expressions easy isro2017

Answer

5.15.6 Regular Expressions: GATE CSE 1996 | Question: 1.8 top https://gateoverflow.in/2712

Which two of the following four regular expressions are equivalent? (ε is the empty string).

i. $(00)^*(\varepsilon+0)$
ii. $(00)^*$
iii. 0^*
iv. $0(00)^*$

A. (i) and (ii)
B. (ii) and (iii)
C. (i) and (iii)
D. (iii) and (iv)

gate1996 theory-of-computation regular-expressions easy

Answer

5.15.7 Regular Expressions: GATE CSE 1997 | Question: 6.4 top https://gateoverflow.in/2260

Which one of the following regular expressions over $\{0,1\}$ denotes the set of all strings not containing 100 as substring?

A. $0^*(1+0)^*$
B. 0^*1010^*
C. $0^*1^*01^*$
D. $0^*(10+1)^*$

gate1997 theory-of-computation regular-expressions normal

Answer

5.15.8 Regular Expressions: GATE CSE 1998 | Question: 1.12 top

https://gateoverflow.in/1649

The string 1101 does not belong to the set represented by

A. $110^*(0+1)$
B. $1(0+1)^*101$
C. $(10)^*(01)^*(00+11)^*$
D. $(00+(11)^*0)^*$

gate1998 theory-of-computation regular-expressions easy multiple-selects

Answer

5.15.9 Regular Expressions: GATE CSE 1998 | Question: 1.9 top

https://gateoverflow.in/1646

If the regular set A is represented by $A=(01+1)^*$ and the regular set B is represented by $B=((01)^*1^*)^*$, which of the following is true?

A. $A \subset B$
B. $B \subset A$
C. A and B are incomparable
D. $A=B$

gate1998 theory-of-computation regular-expressions normal

Answer

5.15.10 Regular Expressions: GATE CSE 1998 | Question: 3b top

https://gateoverflow.in/2941

Give a regular expression for the set of binary strings where every 0 is immediately followed by exactly k 1's and preceded by at least k 1's (k is a fixed integer)

gate1998 theory-of-computation regular-expressions easy descriptive

Answer

5.15.11 Regular Expressions: GATE CSE 2000 | Question: 1.4 top

https://gateoverflow.in/627

Let S and T be languages over $\Sigma=\{a.b\}$ represented by the regular expressions $(a+b^*)^*$ and $(a+b)^*$, respectively. Which of the following is true?

A. $S \subset T$
B. $T \subset S$
C. $S=T$
D. $S \cap T=\phi$

gate2000-cse theory-of-computation regular-expressions easy

Answer

5.15.12 Regular Expressions: GATE CSE 2003 | Question: 14 top

https://gateoverflow.in/905

The regular expression $0^*(10^*)^*$ denotes the same set as

A. $(1^*0)^*1^*$
B. $0+(0+10)^*$
C. $(0+1)^*10(0+1)^*$
D. None of the above

gate2003-cse theory-of-computation regular-expressions easy

Answer

5.15.13 Regular Expressions: GATE CSE 2009 | Question: 15 top https://gateoverflow.in/1307

Which one of the following languages over the alphabet $\{0,1\}$ is described by the regular expression: $(0+1)^*0(0+1)^*0(0+1)^*$?

A. The set of all strings containing the substring 00
B. The set of all strings containing at most two 0's
C. The set of all strings containing at least two 0's
D. The set of all strings that begin and end with either 0 or 1

gate2009-cse theory-of-computation regular-expressions easy

Answer

5.15.14 Regular Expressions: GATE CSE 2010 | Question: 39 top https://gateoverflow.in/2340

Let $L = \{w \in (0+1)^* \mid w \text{ has even number of } 1s\}$. i.e., L is the set of all the bit strings with even numbers of 1s. Which one of the regular expressions below represents L?

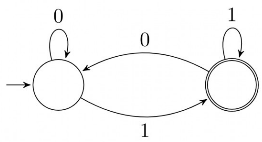
A. $(0^*10^*1)^*$
B. $0^*(10^*10^*)^*$
C. $0^*(10^*1)^*0^*$
D. $0^*1(10^*1)^*10^*$

gate2010-cse theory-of-computation regular-expressions normal

Answer

5.15.15 Regular Expressions: GATE CSE 2014 Set 1 | Question: 36 top https://gateoverflow.in/1914

Which of the regular expressions given below represent the following DFA?



I. $0^*1(1+00^*1)^*$
II. $0^*1^*1+11^*0^*1$
III. $(0+1)^*1$

A. I and II only
B. I and III only
C. II and III only
D. I, II and III

gate2014-cse-set1 theory-of-computation regular-expressions finite-automata easy

Answer

5.15.16 Regular Expressions: GATE CSE 2014 Set 3 | Question: 15 top https://gateoverflow.in/2049

The length of the shortest string NOT in the language (over $\Sigma = \{a, b\}$) of the following regular expression is _______.

$$a^*b^*(ba)^*a^*$$

gate2014-cse-set3 theory-of-computation regular-expressions numerical-answers easy

Answer

5.15.17 Regular Expressions: GATE CSE 2016 Set 1 | Question: 18 top https://gateoverflow.in/39647

Which one of the following regular expressions represents the language: *the set of all binary strings having two consecutive*

$0's$ *and two consecutive*
$1's$?

A. $(0+1)^*0011(0+1)^* + (0+1)^*1100(0+1)^*$
B. $(0+1)^*(00(0+1)^*11 + 11(0+1)^*00)(0+1)^*$
C. $(0+1)^*00(0+1)^* + (0+1)^*11(0+1)^*$
D. $00(0+1)^*11 + 11(0+1)^*00$

gate2016-cse-set1 theory-of-computation regular-expressions normal

Answer

5.15.18 Regular Expressions: GATE CSE 2020 | Question: 7 top https://gateoverflow.in/333224

Which one of the following regular expressions represents the set of all binary strings with an odd number of $1's$?

A. $((0+1)^*1(0+1)^*1)^*10^*$
B. $(0^*10^*10^*)^*0^*1$
C. $10^*(0^*10^*10^*)^*$
D. $(0^*10^*10^*)^*10^*$

gate2020-cse regular-expressions normal theory-of-computation

Answer

5.15.19 Regular Expressions: GATE CSE 2021 Set 2 | Question: 47 top https://gateoverflow.in/357493

Which of the following regular expressions represent(s) the set of all binary numbers that are divisible by three? Assume that the string ϵ is divisible by three.

A. $(0+1(01^*0)^*1)^*$
B. $(0+11+10(1+00)^*01)^*$
C. $(0^*(1(01^*0)^*1)^*)^*$
D. $(0+11+11(1+00)^*00)^*$

gate2021-cse-set2 multiple-selects theory-of-computation regular-expressions

Answer

5.15.20 Regular Expressions: GATE IT 2004 | Question: 7 top https://gateoverflow.in/3648

Which one of the following regular expressions is NOT equivalent to the regular expression $(a+b+c)^*$?

A. $(a^*+b^*+c^*)^*$
B. $(a^*b^*c^*)^*$
C. $((ab)^*+c^*)^*$
D. $(a^*b^*+c^*)^*$

gate2004-it theory-of-computation regular-expressions normal

Answer

5.15.21 Regular Expressions: GATE IT 2005 | Question: 5 top https://gateoverflow.in/3749

Which of the following statements is TRUE about the regular expression 01^*0?

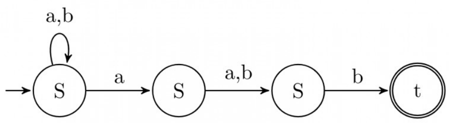
A. It represents a finite set of finite strings.
B. It represents an infinite set of finite strings.
C. It represents a finite set of infinite strings.
D. It represents an infinite set of infinite strings.

gate2005-it theory-of-computation regular-expressions easy

Answer

5.15.22 Regular Expressions: GATE IT 2006 | Question: 5 top https://gateoverflow.in/3544

Which regular expression best describes the language accepted by the non-deterministic automaton below?



A. $(a+b)^*\ a(a+b)b$
B. $(abb)^*$
C. $(a+b)^*\ a(a+b)^*\ b(a+b)^*$
D. $(a+b)^*$

gate2006-it theory-of-computation regular-expressions normal

Answer

5.15.23 Regular Expressions: GATE IT 2007 | Question: 73 top https://gateoverflow.in/3525

Consider the regular expression $R = (a+b)^*\ (aa+bb)\ (a+b)^*$

Which one of the regular expressions given below defines the same language as defined by the regular expression R ?

A. $(a(ba)^* + b(ab)^*)(a+b)^+$
B. $(a(ba)^* + b(ab)^*)^*(a+b)^*$
C. $(a(ba)^*(a+bb) + b(ab)^*(b+aa))(a+b)^*$
D. $(a(ba)^*(a+bb) + b(ab)^*(b+aa))(a+b)^+$

gate2007-it theory-of-computation regular-expressions normal

Answer

5.15.24 Regular Expressions: GATE IT 2008 | Question: 5 top https://gateoverflow.in/3265

Which of the following regular expressions describes the language over $\{0,1\}$ consisting of strings that contain exactly two 1's?

A. $(0+1)^*\ 11(0+1)^*$
B. $0^*\ 110^*$
C. $0^*10^*10^*$
D. $(0+1)^*1(0+1)^*1(0+1)^*$

gate2008-it theory-of-computation regular-expressions easy

Answer

Answers: Regular Expressions

5.15.1 Regular Expressions: GATE CSE 1987 | Question: 10d top https://gateoverflow.in/82455

✔ $0,1,01,10,11,011,110$

$\therefore$ RE : $(0+1+01+10+11+011+110)$

ε and 0110 are not part of RE because of NON-NULL and PROPER substring requirement.

20 votes

-- Akshay Saxena (8.3k points)

5.15.2 Regular Expressions: GATE CSE 1991 | Question: 03,xiii top https://gateoverflow.in/527

✔ Answer is **A** and **C.**

To know the answer let us check all the options:

A. $L(s) \subseteq L(r)$: strings generated by s are any numbers of 1's followed by one 0, i.e., $10, 110, 1110, 1110, \ldots$. Strings

generated by r are is 1 followed by any combination of 0 or 1, i.e., $1, 10, 11, 1110, 101, 110 \dots$. This shows that all the strings that can be generated by s, can also be generated by r it means $L(s) \subseteq L(r)$ is true.
$L(s) \subseteq L(t)$: here strings generated by t are any numbers of 1 (here 1^* means we have strings as $\epsilon, 1, 11, 111, \dots$) followed by only one 0, i.e., $0, 10, 110, 1110, \dots$. So we can see that all the strings that are present in s can also be generated by t, hence $L(s) \subseteq L(t)$ which shows that option **A** is true.

B. $L(r) \subseteq L(s)$: this is false because string 1 which can be generated by r, cannot be generated by s.

C. Same as option **A.**

D. $L(t) \subseteq L(s)$: this is false because string 0 which can be generated by t,cannot be generated by s.

35 votes -- neha pawar (3.3k points)

5.15.3 Regular Expressions: GATE CSE 1992 | Question: 02,xvii top https://gateoverflow.in/575

✔

A. Is the answer
B. RHS generates Σ^* while LHS can't generate strings where r comes after s like sr, srr etc. LHS $\subset$ RHS.
C. LHS generates Σ^* while RHS can't generate strings where r comes after an s. RHS $\subset$ LHS.
D. LHS contains all strings where after an s, no r comes. RHS contains all strings of either r or s but no combination of them. So, RHS $\subset$ LHS.

44 votes -- Arjun Suresh (330k points)

5.15.4 Regular Expressions: GATE CSE 1994 | Question: 2.10 top https://gateoverflow.in/2477

✔ Using Arden's theorem to find the regular expression :

$A = \epsilon + A0 = \epsilon 0^* = 0^*$

$B = A1 + B1 = 0^*1 + B1 = 0^*11^*$

As there are two final states, we should union both RE.

$\implies 0^* + 0^*11^* = 0^*(\epsilon + 11^*) = 0^*(\epsilon + 1^+) = 0^*1^*$

Note that $\epsilon + 1^+ = 1^*$

PS : Ardens theorem - Let P and Q be two regular expressions. If P does not contain null string, then $R = Q + RP$ has a unique solution which is $R = QP^*$

$L = 0^*1^*$
L contains all binary strings where any 1 is not followed by any 0.

47 votes -- Manu Thakur (34.1k points)

5.15.5 Regular Expressions: GATE CSE 1995 | Question: 1.9 , ISRO2017-13 top https://gateoverflow.in/2596

✔ Correct Option: **C**

It has to be started by a letter followed by any number of letters (or) digits.

33 votes -- Gate Keeda (15.9k points)

5.15.6 Regular Expressions: GATE CSE 1996 | Question: 1.8 top https://gateoverflow.in/2712

✔ **Answer is** C.

You can have any no. of 0's as well as null.

A is false because you cannot have single 0 in ii). same for option **B**. In **D** you are forced to have single 0 in iv) whereas not in iii).

26 votes -- Gate Keeda (15.9k points)

5.15.7 Regular Expressions: GATE CSE 1997 | Question: 6.4 top https://gateoverflow.in/2260

✔

> "A regular expression denoting a language (set of strings) means it should generate all string in L and not generate any string not in L"

a. - generates 100
b. doesn't generate 0 (start trying strings in lexicographic order- $0, 1, 00, 01, 10, ...$)
c. doesn't generate 1
d. is the answer

👍 53 votes -- Arjun Suresh (330k points)

5.15.8 Regular Expressions: GATE CSE 1998 | Question: 1.12 top https://gateoverflow.in/1649

✔ Only **(a)** and **(b)** can generate 1101.

In **(c)** after 11, we can not have 01 and so 1101 cannot be generated.

In **(d)** Every 11 followed by 0 and no single occurrence of 1 is possible. So it cannot generate 1101 or 11011.

👍 48 votes -- Arjun Suresh (330k points)

5.15.9 Regular Expressions: GATE CSE 1998 | Question: 1.9 top https://gateoverflow.in/1646

✔ Correct Option: **D**

Both generates all strings over $\{0, 1\}$ where a 0 is immediately followed by a 1.

👍 31 votes -- Arjun Suresh (330k points)

5.15.10 Regular Expressions: GATE CSE 1998 | Question: 3b top https://gateoverflow.in/2941

✔ $1^*1^k(01^k)^* + 1^*$

This is correct expression, this considering chance of not having any 0's (In that case string can also be empty string).

PS: The question assumes a fixed k, and we should expand the given expression for that k to get a valid regular expression.

👍 41 votes -- Akash Kanase (36k points)

5.15.11 Regular Expressions: GATE CSE 2000 | Question: 1.4 top https://gateoverflow.in/627

✔ Correct Option: **C**

$S = T$. Both generates all strings over Σ.

👍 30 votes -- Arjun Suresh (330k points)

5.15.12 Regular Expressions: GATE CSE 2003 | Question: 14 top https://gateoverflow.in/905

✔

A. Is the answer. Both (A) and the given expression generates all strings over Σ.
B. doesn't generate 11
C. doesn't generate 11

👍 41 votes -- Arjun Suresh (330k points)

5.15.13 Regular Expressions: GATE CSE 2009 | Question: 15 top https://gateoverflow.in/1307

✔ Correct Option: **C**

Counter example for other choices:

A. 1010 is accepted which doesn't contain 00

B. 000 is accepted
C. is the answer.
D. 01 is not accepted

👍 33 votes -- Arjun Suresh (330k points)

5.15.14 Regular Expressions: GATE CSE 2010 | Question: 39 top ☛ https://gateoverflow.in/2340

✔

A. - If the string contains a 1, it must end in a 1 hence cannot generate all bit strings with even number of 1's (eg, 1010)
B. - is the answer
C. - between the second and third 1's a 0 is not allowed (eg, 011011)
D. - 00 is not allowed, zero is an even number.

👍 65 votes -- Arjun Suresh (330k points)

5.15.15 Regular Expressions: GATE CSE 2014 Set 1 | Question: 36 top ☛ https://gateoverflow.in/1914

✔ Correct Option: **B**

(II) doesn't generate 11011 which is accepted by the given **DFA**.

👍 56 votes -- Arjun Suresh (330k points)

5.15.16 Regular Expressions: GATE CSE 2014 Set 3 | Question: 15 top ☛ https://gateoverflow.in/2049

✔ $R = a^*b^*(ba)^*a^*$

for finding shortest string that is not in language it is better to look strings of length 0, then of length 1 and so on

length0$\{\epsilon\}$ is in L

length1$\{a, b\}$ all belong to L

length2$\{aa, ab, ba, bb\}$ all belong to L

length3$\{aaa, aab, aba, abb, baa, bab, bba, bbb\}$ ***bab does not belong to L***

👍 84 votes -- Praveen Saini (41.9k points)

5.15.17 Regular Expressions: GATE CSE 2016 Set 1 | Question: 18 top ☛ https://gateoverflow.in/39647

✔ Set of all binary strings having two consecutive 0s and two consecutive 1s

$Anything\ 00\ Anything\ 11 Anything\ + Anything\ 11\ Anything\ 00\ Anything$

$(0+1)^*00(0+1)^*11(0+1)^* + (0+1)^*11(0+1)^*00(0+1)^*$

And it is same after taking common.

$(0+1)^*(00(0+1)^*11 + 11(0+1)^*00)(0+1)^*$

So, **option B** is answer.

Neither they said Both are immediate nor they give a predefined order, so it should be as above

👍 105 votes -- Praveen Saini (41.9k points)

5.15.18 Regular Expressions: GATE CSE 2020 | Question: 7 top ☛ https://gateoverflow.in/333224

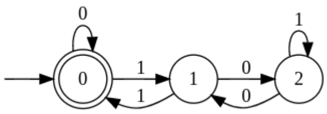
✔ Regular expression in option A cannot generate 001
Regular expression in option B cannot generate 100
Regular expression in option C cannot generate 001
Regular expression in option D cannot generate 001

Hence, mark was given to everyone in GATE for this question.

👍 23 votes -- Aditya Pradhan (625 points)

5.15.19 Regular Expressions: GATE CSE 2021 Set 2 | Question: 47 top ☛ https://gateoverflow.in/357493

✔



The above is the minimal DFA for the given language.

From the given DFA we can see all option but options except D are correct.

5 votes -- zxy123 (2.5k points)

5.15.20 Regular Expressions: GATE IT 2004 | Question: 7 top https://gateoverflow.in/3648

✔ A) $(a^* + b^* + c^*)^* = (\wedge + a + aa + \ldots + b + bb + \ldots$
$+ c + cc \ldots)^* = (a + b + c + aa + .. + bb + .. + cc$
$+ ..)^* = (a + b + c)^*$
[any combination of rest of aa, bb, cc, etc already come in $(a + b + c)^*$]

B) $(a^* b^* c^*)^* = (a^* + b^* + c^* + a^* b^* + b^* c^* + a^* c^*$
$+ ..)^* = (a + b + c + \ldots)^* = (a + b + c)^*$

C)
$((ab)^* + c^*)^* = (ab + c + \wedge + abab + \ldots)^*$
$= (ab + c)^*$

D) $(a^* b^* + c^*)^* = (a^* + b^* + c^* + \ldots)^*$
$= (a + b + c + ..)^* = (a + b + c)^*$

Correct Answer: C

37 votes -- Praveen Saini (41.9k points)

5.15.21 Regular Expressions: GATE IT 2005 | Question: 5 top https://gateoverflow.in/3749

✔ Correct Option: **B**

Infinite set (because of *) of finite strings. A string is defined as a **FINITE sequence** of characters and hence can never be infinite.

48 votes -- Arjun Suresh (330k points)

5.15.22 Regular Expressions: GATE IT 2006 | Question: 5 top https://gateoverflow.in/3544

✔ **A** is the answer.

Say s_1, s_2, s_3 and t are the states (in sequence) with s_1 being the start state

$s_1 = \epsilon + s_1 a + s_1 b = \epsilon + s_1(a + b) = (a + b)^*$ [using Arden's theorem $R = Q + RP$, then $R = QP^*$][ϵ because of being the start state]

$s_2 = s_1 a = (a + b)^* a$

$s_3 = s_2 a + s_2 b = s_2(a + b) = (a + b)^* a(a + b)$

$t = s_3 b = (a + b)^* a(a + b)b$

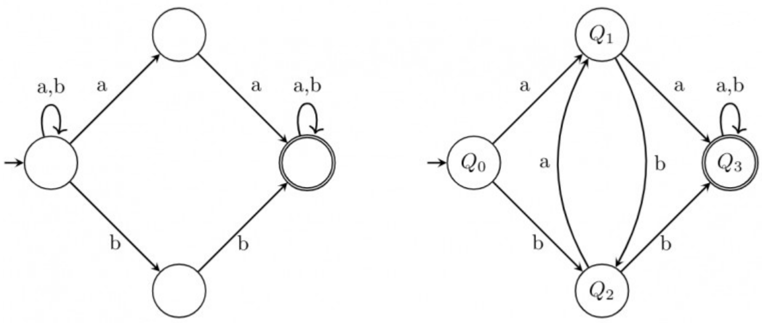
t is final state so regular expression is $(a + b)^* a(a + b)b$

35 votes -- Praveen Saini (41.9k points)

5.15.23 Regular Expressions: GATE IT 2007 | Question: 73 top https://gateoverflow.in/3525

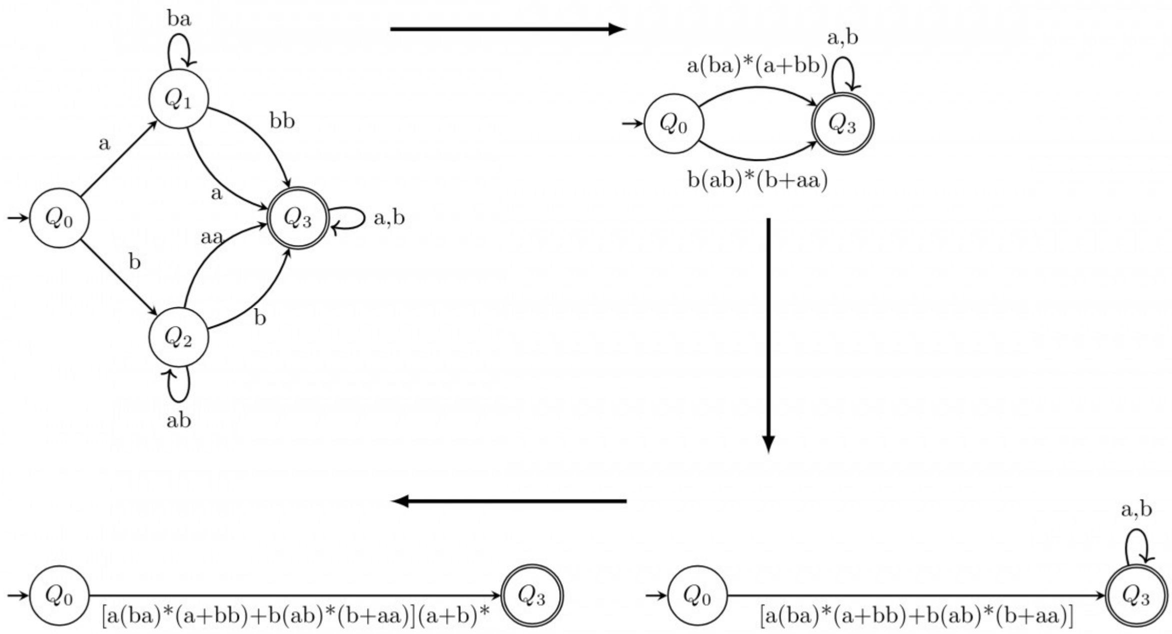
✔ $R = (a + b)^*(aa + bb)(a + b)^*$

Having,



NFA Equivalent DFA

Which is equivalent to the following Transition graph [by removing transition from Q_1 to Q_2 and Q_2 to Q_1 but does not affect the accepted language, be careful] and can be converted to an equivalent regular expression as shown below.



So, equivalent regular expression is $[a(ba)^*(a+bb)+b(ab)^*(b+aa)](a+b)^*$

Option **C** is answer.

👍 53 votes -- Praveen Saini (41.9k points)

Another quick approach of solving this question for keen observers :-

Observe that aa or bb is minimal string that is possible in first Regular Expression $(a+b)^*(aa+bb)(a+b)^*$.

(A) We can have ba or ab as minimal strings which is not possible in $(a+b)^*(aa+bb)(a+b)^*$

(B) We can have empty string, which is not possible in $(a+b)^*(aa+bb)(a+b)^*$.

(D) Minimum string length is 3, aa or bb is not possible in this RE.

This rules out options A, B and D. So, option C must be the answer.

👍 76 votes -- Akash Kanase (36k points)

5.15.24 Regular Expressions: GATE IT 2008 | Question: 5 top

https://gateoverflow.in/3265

✔

A. with at least 2 consecutive 1's, any no of 0's and any no of 1's
B. exactly two consecutive 1's
C. exactly two 1's but need not be consecutive
D. any no of 1's and 0's with at least two 1's

Hence, (**C**) is the correct option.

33 votes -- Manu Thakur (34.1k points)

5.16 Regular Grammar (3) top

5.16.1 Regular Grammar: GATE CSE 1990 | Question: 15a top https://gateoverflow.in/86864

Is the language generated by the grammar G regular? If so, give a regular expression for it, else prove otherwise

- G:
  - $S \rightarrow aB$
  - $B \rightarrow bC$
  - $C \rightarrow xB$
  - $C \rightarrow c$

gate1990 descriptive theory-of-computation regular-languages regular-grammar

Answer

5.16.2 Regular Grammar: GATE CSE 2015 Set 2 | Question: 35 top https://gateoverflow.in/8159

Consider the alphabet $\Sigma = \{0, 1\}$, the null/empty string λ and the set of strings $X_0, X_1,$ and X_2 generated by the corresponding non-terminals of a regular grammar. $X_0, X_1,$ and X_2 are related as follows.

- $X_0 = 1X_1$
- $X_1 = 0X_1 + 1X_2$
- $X_2 = 0X_1 + \{\lambda\}$

Which one of the following choices precisely represents the strings in X_0?

A. $10(0^* + (10)^*)1$
B. $10(0^* + (10)^*)^*1$
C. $1(0 + 10)^*1$
D. $10(0 + 10)^*1 + 110(0 + 10)^*1$

gate2015-cse-set2 theory-of-computation regular-grammar normal

Answer

5.16.3 Regular Grammar: GATE IT 2006 | Question: 29 top https://gateoverflow.in/3568

Consider the regular grammar below

$S \rightarrow bS \mid aA \mid \epsilon$
$A \rightarrow aS \mid bA$

The Myhill-Nerode equivalence classes for the language generated by the grammar are

A. $\{w \in (a + b)^* \mid \#a(w) \text{ is even}\}$ and $\{w \in (a + b)^* \mid \#a(w) \text{ is odd}\}$
B. $\{w \in (a + b)^* \mid \#a(w) \text{ is even}\}$ and $\{w \in (a + b)^* \mid \#b(w) \text{ is odd}\}$
C. $\{w \in (a + b)^* \mid \#a(w) = \#b(w)\}$ and $\{w \in (a + b)^* \mid \#a(w) \neq \#b(w)\}$
D. $\{\epsilon\}, \{wa \mid w \in (a + b)^*\}$ and $\{wb \mid w \in (a + b)^*\}$

gate2006-it theory-of-computation normal regular-grammar

Answer

Answers: Regular Grammar

5.16.1 Regular Grammar: GATE CSE 1990 | Question: 15a top https://gateoverflow.in/86864

✔ First of all this grammar is right linear. And we know:

> Two special types of linear grammars are the following:
>
> - the **left-linear** or left regular grammars, in which **all nonterminals** in right hand sides are **at the left ends**;
> - the **right-linear** or right regular grammars, in which **all nonterminals** in right hand sides are **at the right ends**.

Hence the given grammar is regular and hence the language generated by regular grammar will also be regular.
Alternatively we can also write a regular expression for it. Let us see how to do it:

Given grammar:

$G: \ S \to aB$

$B \to bC$

$C \to xB \mid c$

So, we can reverse substitution from C onwards to S to see what S generates..

Substituting the yield of C in B, we have:

$B \to bxB \mid bc$

which gives $B = (bx)^{*}bc$

Now, substituting B in S we have:

$\mathbf{S \to aB}$

$\mathbf{S = a(bx)^{*}bc}$

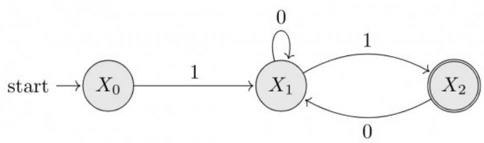
Hence, the corresponding regular expression is : $\mathbf{a(bx)^{*}bc}$

References

👍 31 votes -- HABIB MOHAMMAD KHAN (67.5k points)

5.16.2 Regular Grammar: GATE CSE 2015 Set 2 | Question: 35 top ☛ https://gateoverflow.in/8159

✔



Convert the given transitions to a state diagram

From the given diagram we can write

$X_0 = 1(0 + 10)^{*}1$

Correct Option: **C**

👍 101 votes -- Umang Raman (12.2k points)

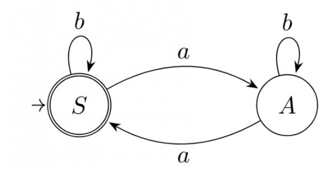
5.16.3 Regular Grammar: GATE IT 2006 | Question: 29 top ☛ https://gateoverflow.in/3568

✔ Option A is the correct answer.

Before doing this question we should know the following points.

1. Number of equivalence classes = Number of states in Minimal FA (MFA)
2. In MFA, we get some language at each and every stage. These languages are mutually exclusive and are called equivalence classes.

So to solve this question, first convert the given Right Linear Regular Grammar into DFA.



There are two states named S and A.

The language at state S represents one Equivalence Class. $\left\{w \in (a+b)^* \mid \#\text{a(w) is even}\right\}$

The language at State A represents another Equivalence Class. $\left\{w \in (a+b)^* \mid \#\text{a(w) is odd}\right\}$

111 votes -- Rajesh Pradhan (18.9k points)

5.17 Regular Languages (32) top

5.17.1 Regular Languages: GATE CSE 1987 | Question: 2h top

https://gateoverflow.in/80589

State whether the following statements are TRUE or FALSE:

Regularity is preserved under the operation of string reversal.

gate1987 theory-of-computation regular-languages true-false

Answer

5.17.2 Regular Languages: GATE CSE 1987 | Question: 2i top

https://gateoverflow.in/80590

State whether the following statements are TRUE or FALSE:

All subsets of regular sets are regular.

gate1987 theory-of-computation regular-languages true-false

Answer

5.17.3 Regular Languages: GATE CSE 1990 | Question: 3-viii top

https://gateoverflow.in/84837

Let R_1 and R_2 be regular sets defined over the alphabet Σ Then:

A. $R_1 \cap R_2$ is not regular.
B. $R_1 \cup R_2$ is regular.
C. $\Sigma^* - R_1$ is regular.
D. R_1^* is not regular.

gate1990 normal theory-of-computation regular-languages multiple-selects

Answer

5.17.4 Regular Languages: GATE CSE 1991 | Question: 03,xiv top

https://gateoverflow.in/528

Which of the following is the strongest correct statement about a finite language over some finite alphabet Σ?

A. It could be undecidable
B. It is Turing-machine recognizable
C. It is a context-sensitive language.
D. It is a regular language.
E. None of the above,

gate1991 theory-of-computation easy regular-languages multiple-selects

Answer

5.17.5 Regular Languages: GATE CSE 1995 | Question: 2.24 top

https://gateoverflow.in/2637

Let $\Sigma = \{0,1\}, L = \Sigma^*$ and $R = \{0^n 1^n \mid n > 0\}$ then the languages $L \cup R$ and R are respectively

A. regular, regular

B. not regular, regular
C. regular, not regular
D. not regular, not regular

gate1995 theory-of-computation easy regular-languages

Answer

5.17.6 Regular Languages: GATE CSE 1996 | Question: 1.10 top

https://gateoverflow.in/2714

Let $L \subseteq \Sigma^*$ where $\Sigma = \{a, b\}$. Which of the following is true?

a. $L = \{x \mid x$ has an equal number of a's and b's$\}$ is regular
b. $L = \{a^n b^n \mid n \geq 1\}$ is regular
c. $L = \{x \mid x$ has more number of a's than b's$\}$ is regular
d. $L = \{a^m b^n \mid m \geq 1, n \geq 1\}$ is regular

gate1996 theory-of-computation normal regular-languages

Answer

5.17.7 Regular Languages: GATE CSE 1998 | Question: 2.6 top

https://gateoverflow.in/1678

Which of the following statements is false?

a. Every finite subset of a non-regular set is regular
b. Every subset of a regular set is regular
c. Every finite subset of a regular set is regular
d. The intersection of two regular sets is regular

gate1998 theory-of-computation easy regular-languages

Answer

5.17.8 Regular Languages: GATE CSE 1999 | Question: 6 top

https://gateoverflow.in/1505

A. Given that A is regular and $(A \cup B)$ is regular, does it follow that B is necessarily regular? Justify your answer.
B. Given two finite automata $M1, M2$, outline an algorithm to decide if $L(M1) \subset L(M2)$. (note: strict subset)

gate1999 theory-of-computation normal regular-languages descriptive

Answer

5.17.9 Regular Languages: GATE CSE 2000 | Question: 2.8 top

https://gateoverflow.in/655

What can be said about a regular language L over $\{a\}$ whose minimal finite state automaton has two states?

A. L must be $\{a^n \mid n$ is odd$\}$
B. L must be $\{a^n \mid n$ is even$\}$
C. L must be $\{a^n \mid n \geq 0\}$
D. Either L must be $\{a^n \mid n$ is odd$\}$, or L must be $\{a^n \mid n$ is even$\}$

gate2000-cse theory-of-computation easy regular-languages

Answer

5.17.10 Regular Languages: GATE CSE 2001 | Question: 1.4 top

https://gateoverflow.in/697

Consider the following two statements:

$S_1 : \{0^{2n} \mid n \geq 1\}$ is a regular language

$S_2 : \{0^m 1^n 0^{m+n} \mid m \geq 1 \text{ and } n \geq 1\}$ is a regular language

Which of the following statement is correct?

A. Only S_1 is correct
B. Only S_2 is correct
C. Both S_1 and S_2 are correct
D. None of S_1 and S_2 is correct

gate2001-cse theory-of-computation easy regular-languages

Answer

5.17.11 Regular Languages: GATE CSE 2001 | Question: 2.6 top https://gateoverflow.in/724

Consider the following languages:

- $L1 = \{ww \mid w \in \{a,b\}^*\}$
- $L2 = \left\{ww^R \mid w \in \{a,b\}^*, w^R \text{ is the reverse of w}\right\}$
- $L3 = \left\{0^{2i} \mid \text{i is an integer}\right\}$
- $L4 = \left\{0^{i^2} \mid \text{i is an integer}\right\}$

Which of the languages are regular?

A. Only $L1$ and $L2$
B. Only $L2, L3$ and $L4$
C. Only $L3$ and $L4$
D. Only $L3$

gate2001-cse theory-of-computation normal regular-languages

Answer

5.17.12 Regular Languages: GATE CSE 2006 | Question: 29 top https://gateoverflow.in/992

If s is a string over $(0+1)^*$ then let $n_0(s)$ denote the number of 0's in s and $n_1(s)$ the number of 1's in s. Which one of the following languages is not regular?

A. $L = \{s \in (0+1)^* \mid n_0(s) \text{ is a 3-digit prime }\}$
B. $L = \{s \in (0+1)^* \mid \text{ for every prefix s' of s, } \mid n_0(s') - n_1(s') \mid \leq 2\}$
C. $L = \{s \in (0+1)^* \mid n_0(s) - n_1(s) \mid \leq 4\}$
D. $L = \{s \in (0+1)^* \mid n_0(s) \mod 7 = n_1(s) \mod 5 = 0\}$

gate2006-cse theory-of-computation normal regular-languages

Answer

5.17.13 Regular Languages: GATE CSE 2007 | Question: 31 top https://gateoverflow.in/1229

Which of the following languages is regular?

A. $\{ww^R \mid w \in \{0,1\}^+\}$
B. $\{ww^Rx \mid x, w \in \{0,1\}^+\}$
C. $\{wxw^R \mid x, w \in \{0,1\}^+\}$
D. $\{xww^R \mid x, w \in \{0,1\}^+\}$

gate2007-cse theory-of-computation normal regular-languages

Answer

5.17.14 Regular Languages: GATE CSE 2007 | Question: 7 top https://gateoverflow.in/1205

Which of the following is TRUE?

A. Every subset of a regular set is regular

B. Every finite subset of a non-regular set is regular
C. The union of two non-regular sets is not regular
D. Infinite union of finite sets is regular

gate2007-cse theory-of-computation easy regular-languages

Answer

5.17.15 Regular Languages: GATE CSE 2008 | Question: 53 top

https://gateoverflow.in/476

Which of the following are regular sets?

I. $\{a^n b^{2m} \mid n \geq 0, m \geq 0\}$
II. $\{a^n b^m \mid n = 2m\}$
III. $\{a^n b^m \mid n \neq m\}$
IV. $\{xcy \mid x, y, \in \{a, b\}^*\}$

A. I and IV only
B. I and III only
C. I only
D. IV only

gate2008-cse theory-of-computation normal regular-languages

Answer

5.17.16 Regular Languages: GATE CSE 2011 | Question: 24 top

https://gateoverflow.in/3429

Let P be a regular language and Q be a context-free language such that $Q \subseteq P$. (For example, let P be the language represented by the regular expression p^*q^* and Q be $\{p^n q^n \mid n \in N\}$). Then which of the following is **ALWAYS** regular?

A. $P \cap Q$
B. $P - Q$
C. $\Sigma^* - P$
D. $\Sigma^* - Q$

gate2011-cse theory-of-computation easy regular-languages

Answer

5.17.17 Regular Languages: GATE CSE 2012 | Question: 25 top

https://gateoverflow.in/1609

Given the language $L = \{ab, aa, baa\}$, which of the following strings are in L^*?

1. $abaabaaabaa$
2. $aaaabaaaa$
3. $baaaaabaaaab$
4. $baaaaabaa$

A. 1, 2 and 3
B. 2, 3 and 4
C. 1, 2 and 4
D. 1, 3 and 4

gate2012-cse theory-of-computation easy regular-languages

Answer

5.17.18 Regular Languages: GATE CSE 2013 | Question: 8 top

https://gateoverflow.in/1417

Consider the languages $L_1 = \phi$ and $L_2 = \{a\}$. Which one of the following represents $L_1 {L_2}^* \cup {L_1}^*$?

A. $\{\epsilon\}$
B. ϕ
C. a^*

D. $\{\epsilon, a\}$

gate2013-cse theory-of-computation normal regular-languages

Answer

5.17.19 Regular Languages: GATE CSE 2014 Set 1 | Question: 15 top https://gateoverflow.in/1781

Which one of the following is **TRUE**?

A. The language $L = \{a^n b^n \mid n \geq 0\}$ is regular.
B. The language $L = \{a^n \mid n \text{ is prime }\}$ is regular.
C. The language L= $\{w \mid w \text{ has } 3k+1 \; b's \text{ for some } k \in N \text{ with } \Sigma = \{a, b\}\}$ is regular.
D. The language $L = \{ww \mid w \in \Sigma^* \text{ with } \Sigma = \{0, 1\}\}$ is regular.

gate2014-cse-set1 theory-of-computation regular-languages normal

Answer

5.17.20 Regular Languages: GATE CSE 2014 Set 2 | Question: 15 top https://gateoverflow.in/1971

If $L_1 = \{a^n \mid n \geq 0\}$ and $L_2 = \{b^n \mid n \geq 0\}$, consider

I. $L_1 . L_2$ is a regular language
II. $L_1 . L_2 = \{a^n b^n \mid n \geq 0\}$

Which one of the following is CORRECT?

A. Only I
B. Only II
C. Both I and II
D. Neither I nor II

gate2014-cse-set2 theory-of-computation normal regular-languages

Answer

5.17.21 Regular Languages: GATE CSE 2014 Set 2 | Question: 36 top https://gateoverflow.in/1995

Let $L_1 = \{w \in \{0, 1\}^* \mid w$ has at least as many occurrences of $(110)'$ s as $(011)'$ s$\}$. Let $L_2 = \{w \in \{0, 1\}^* \mid w$ has at least as many occurrences of $(000)'$ s as $(111)'$ s$\}$. Which one of the following is TRUE?

A. L_1 is regular but not L_2
B. L_2 is regular but not L_1
C. Both L_1 and L_2 are regular
D. Neither L_1 nor L_2 are regular

gate2014-cse-set2 theory-of-computation normal regular-languages

Answer

5.17.22 Regular Languages: GATE CSE 2015 Set 2 | Question: 51 top https://gateoverflow.in/8254

Which of the following is/are regular languages?

$L_1 : \left\{wxw^R \mid w, x \in \{a, b\}^* \text{ and } |w|, |x| > 0\right\}, w^R$ is the reverse of string w

$L_2 : \{a^n b^m \mid m \neq n \text{ and } m, n \geq 0\}$

$L_3 : \{a^p b^q c^r \mid p, q, r \geq 0\}$

A. L_1 and L_3 only
B. L_2 only
C. L_2 and L_3 only
D. L_3 only

gate2015-cse-set2 theory-of-computation normal regular-languages

Answer

5.17.23 Regular Languages: GATE CSE 2016 Set 2 | Question: 17 top

https://gateoverflow.in/39542

Language L_1 is defined by the grammar: $S_1 \rightarrow aS_1b \mid \varepsilon$

Language L_2 is defined by the grammar: $S_2 \rightarrow abS_2 \mid \varepsilon$

Consider the following statements:

- P: L_1 is regular
- Q: L_2 is regular

Which one of the following is **TRUE**?

A. Both P and Q are true.
B. P is true and Q is false.
C. P is false and Q is true.
D. Both P and Q are false.

gate2016-cse-set2 theory-of-computation normal regular-languages

Answer

5.17.24 Regular Languages: GATE CSE 2018 | Question: 52 top

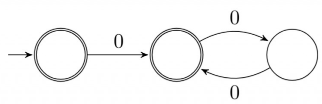
https://gateoverflow.in/204127

Given a language L, define L^i as follows:

$$L^0 = \{\varepsilon\}$$

$$L^i = L^{i-1} \bullet L \text{ for all } I > 0$$

The order of a language L is defined as the smallest k such that $L^k = L^{k+1}$. Consider the language L_1 (over alphabet O) accepted by the following automaton.



The order of L_1 is ____

gate2018-cse theory-of-computation numerical-answers regular-languages

Answer

5.17.25 Regular Languages: GATE CSE 2019 | Question: 7 top

https://gateoverflow.in/302841

If L is a regular language over $\Sigma = \{a, b\}$, which one of the following languages is NOT regular?

A. $L.L^R = \{xy \mid x \in L, y^R \in L\}$
B. $\{ww^R \mid w \in L\}$
C. Prefix $(L) = \{x \in \Sigma^* \mid \exists y \in \Sigma^* \text{ such that } xy \in L\}$
D. Suffix $(L) = \{y \in \Sigma^* \mid \exists x \in \Sigma^* \text{ such that } xy \in L\}$

gate2019-cse theory-of-computation regular-languages

Answer

5.17.26 Regular Languages: GATE CSE 2020 | Question: 51 top

https://gateoverflow.in/333180

Consider the following language.

$L = \{x \in \{a, b\}^* \mid \text{number of } a\text{'s in } x \text{ divisible by } 2 \text{ but not divisible by } 3\}$

The minimum number of states in DFA that accepts L is _________

gate2020-cse numerical-answers theory-of-computation regular-languages

Answer

5.17.27 Regular Languages: GATE CSE 2020 | Question: 8 top

https://gateoverflow.in/333223

Consider the following statements.

I. If $L_1 \cup L_2$ is regular, then both L_1 and L_2 must be regular.
II. The class of regular languages is closed under infinite union.

Which of the above statements is/are TRUE?

A. I only
B. II only
C. Both I and II
D. Neither I nor II

gate2020-cse theory-of-computation regular-languages

Answer

5.17.28 Regular Languages: GATE CSE 2021 Set 2 | Question: 9 top

https://gateoverflow.in/357531

Let $L \subseteq \{0,1\}^*$ be an arbitrary regular language accepted by a minimal DFA with k states. Which one of the following languages must necessarily be accepted by a minimal DFA with k states?

A. $L - \{01\}$
B. $L \cup \{01\}$
C. $\{0,1\}^* - L$
D. $L \cdot L$

gate2021-cse-set2 theory-of-computation finite-automata regular-languages

Answer

5.17.29 Regular Languages: GATE IT 2006 | Question: 30 top

https://gateoverflow.in/3569

Which of the following statements about regular languages is NOT true ?

A. Every language has a regular superset
B. Every language has a regular subset
C. Every subset of a regular language is regular
D. Every subset of a finite language is regular

gate2006-it theory-of-computation easy regular-languages

Answer

5.17.30 Regular Languages: GATE IT 2006 | Question: 80 top

https://gateoverflow.in/3624

Let L be a regular language. Consider the constructions on L below:

I. repeat $(L) = \{ww \mid w \in L\}$
II. prefix $(L) = \{u \mid \exists v : uv \in L\}$
III. suffix $(L) = \{v \mid \exists u : uv \in L\}$
IV. half $(L) = \{u \mid \exists v : |v| = |u| \text{ and } uv \in L\}$

Which of the constructions could lead to a non-regular language?

A. Both I and IV
B. Only I
C. Only IV
D. Both II and III

gate2006-it theory-of-computation normal regular-languages

Answer

5.17.31 Regular Languages: GATE IT 2006 | Question: 81 top

https://gateoverflow.in/3637

Let L be a regular language. Consider the constructions on L below:

I. $\text{repeat}(L) = \{ww \mid w \in L\}$
II. $\text{prefix}(L) = \{u \mid \exists v : uv \in L\}$
III. $\text{suffix}(L) = \{v \mid \exists u : uv \in L\}$
IV. $\text{half}(L) = \{u \mid \exists v : |v| = |u| \text{ and } uv \in L\}$

Which of the constructions could lead to a non-regular language?

a. Both I and IV
b. Only 1
c. Only IV
d. Both II and III

Which choice of L is best suited to support your answer above?

A. $(a+b)^*$
B. $\{\epsilon, a, ab, bab\}$
C. $(ab)^*$
D. $\{a^n b^n \mid n \geq 0\}$

gate2006-it theory-of-computation normal regular-languages

Answer

5.17.32 Regular Languages: GATE IT 2008 | Question: 35 top

https://gateoverflow.in/3345

Which of the following languages is (are) non-regular?

- $L_1 = \{0^m 1^n \mid 0 \leq m \leq n \leq 10000\}$
- $L_2 = \{w \mid w \text{ reads the same forward and backward}\}$
- $L_3 = \{w \in \{0,1\}^* \mid w \text{ contains an even number of 0's and an even number of 1's }\}$

A. L_2 and L_3 only
B. L_1 and L_2 only
C. L_3 only
D. L_2 only

gate2008-it theory-of-computation normal regular-languages

Answer

Answers: Regular Languages

5.17.1 Regular Languages: GATE CSE 1987 | Question: 2h top

https://gateoverflow.in/80589

✔ **True**!
Reverse all the edges and interchange final and initial states in **DFA.**

18 votes

-- Motamarri Anusha (8.6k points)

5.17.2 Regular Languages: GATE CSE 1987 | Question: 2i top

https://gateoverflow.in/80590

✔ **False**!
a^*b^* is regular but its subset $a^n b^n$ is not regular.

20 votes

-- Motamarri Anusha (8.6k points)

5.17.3 Regular Languages: GATE CSE 1990 | Question: 3-viii top

https://gateoverflow.in/84837

✔ Regular Languages are closed under

1. Intersection
2. Union
3. Complement

4. Kleen-Closure

$\Sigma^* - R_1$ is the complement of R_1

Correct Options: **B;C**

26 votes -- Akhil Nadh PC (16.5k points)

5.17.4 Regular Languages: GATE CSE 1991 | Question: 03,xiv top https://gateoverflow.in/528

(**B**), (**C**) and (**D**) are true. But the strongest answer would be (D) a regular language. It is trivial to say that a finite set of strings (finite language) can be accepted using a finite set of states. And regular language $\subset$ context-free $\subset$ context-sensitive $\subset$ Turing recognizable, would imply that regular language is the strongest answer.

29 votes -- gatecse (62.6k points)

5.17.5 Regular Languages: GATE CSE 1995 | Question: 2.24 top https://gateoverflow.in/2637

Answer is (**C**). $L \cup R$ is nothing but L as R is a subset of L and hence regular. R is deterministic context-free but not regular as we require a stack to keep the count of 0's to match that of 1's.

24 votes -- Arjun Suresh (330k points)

5.17.6 Regular Languages: GATE CSE 1996 | Question: 1.10 top https://gateoverflow.in/2714

Correct Option: **D**

Since n and m are independent finite memory suffices.

Options (a) and (b) are the same. They and option (c) require keeping track of the counts of $a's$ which cannot be done using a finite automata but can be done using a DPDA and hence are not regular but DCFL.

13 votes -- Gate Keeda (15.9k points)

5.17.7 Regular Languages: GATE CSE 1998 | Question: 2.6 top https://gateoverflow.in/1678

Correct Option: **B**

Any language is a subset of Σ^* which is a regular set. So, if we take any non-regular language, it is a subset of a regular language.

(a) and **(c)** are regular as any finite language is regular.

(d) is regular as regular set is closed under intersection.

23 votes -- Arjun Suresh (330k points)

5.17.8 Regular Languages: GATE CSE 1999 | Question: 6 top https://gateoverflow.in/1505

a. A is regular , $A \cup B$ is regular , then B is not necessary regular
   example :- $A = (a+b)^*$ $B = a^n b^n \; n \geq 0$ $A \cup B$ is $(a+b)^*$ while B is not regular.

b. We have two machine $M1$ and $M2$
   draw a DFA using $M1$ and $M2$ where start state is, say, $p0q0$ (where $p0$ is start state in $M1$ and $q0$ is start state in $M2$)
   $\partial(p0q0, 0) = \partial(p0, 0) \cup \partial(q0, 0)$
   if $L(M1) \subseteq L(M2)$
   Then final state of $M1$ will come together with final state of $M2$, while Final state of $M2$ can come alone.
   i.e. all inputs of $M1$ is also in machine $M2$, and there may be different inputs in $M2$.

38 votes -- Praveen Saini (41.9k points)

5.17.9 Regular Languages: GATE CSE 2000 | Question: 2.8 top https://gateoverflow.in/655

The question is not precise here. Since, minimal DFA has two states exactly one must be final. Because if

1. No state is final, no strings can be accepted and for this only one state is required in the minimal DFA
2. If both the states are final, then they can be merged to a single state and hence it won't be a minimal DFA

Now, with one final state and two states

1. if we make a transition on first a to final state and stay there for any remaining number of $a's$, the language we get is $L = \{a^n \mid n > 0\}$ which is $a^* - \{\epsilon\}$
2. Like in above we do the transition must if the initial state is made final, then the only string accepted is ϵ

The above two cases are ignored in the given options.

The remaining possibility is for each input symbol a, the DFA transitions between the first and second states. Then,

1. if initial state is final we get $L = \{a^n \mid n \text{ is even}\}$
2. if initial state is not final we get $L = \{a^n \mid n \text{ is odd}\}$

So, none of the options is correct though D is the best option to pick.

1 votes -- Arjun Suresh (330k points)

5.17.10 Regular Languages: GATE CSE 2001 | Question: 1.4 top https://gateoverflow.in/697

✔ Only S_1 is correct!

A **DFA** with 3 states will be needed, as the strings in the language S_1 are $00, 0000, 000000$, and so on. which is the set of all strings with even number of $0's$ and with length greater than 0. We would have needed only 2 sates had empty string also been in the language but $n \geq 1$ prohibits it and so we need 3 states in our DFA. This assumes that the language is over $\{0\}$ and not $\{0, 1\}$.

S_2 is **DCFL** as we need to do infinite counting of $0's$ and $1's$ here.

22 votes -- Bhagirathi Nayak (11.7k points)

5.17.11 Regular Languages: GATE CSE 2001 | Question: 2.6 top https://gateoverflow.in/724

✔ $L1 = \{ww \mid w \in \{a, b\}^*\}$ CSL

$L2 = \{ww^R \mid w \in \{a, b\}^*, w^R \text{ is the reverse of w}\}$ Palindrome, so CFL

$L3 = \{0^{2i} \mid i \text{ is an integer}\}$ Linear power and regular expression can be stated as $(00)^*$

$L4 = \{0^{i^2} \mid i \text{ is an integer}\}$ Non-linear power, so CSL

Therefore, answer is option D.

20 votes -- Umang Raman (12.2k points)

5.17.12 Regular Languages: GATE CSE 2006 | Question: 29 top https://gateoverflow.in/992

✔ A. There are only finite 3 digit primes. Let the largest of them be X. Now we need $X + 2$ states in our DFA (including one for $count = 0$ and one for $count > X$). We do not need a change of state for any 1.

B. Here we need just 6 states to recognise L.

1. $\#0 - \#1 = 0$
2. $\#0 - \#1 = 1$
3. $\#0 - \#1 = 2$
4. $\#0 - \#1 = -1$
5. $\#0 - \#1 = -2$

If the difference goes above 2 or below -2, we go to a dead state. All other states are accepting. This transition to dead state is possible because of the words "for every prefix s' of s" in L and that is what makes this language regular.

C. L is not regular

$\#0 - \#1 = 1$
$\#0 - \#1 = 2$
$\#0 - \#1 = 3$
$\#0 - \#1 = 4$

$\#0 - \#1 = 5$
$\vdots$
$\#0 - \#1 = 1000$
$\vdots$

All these form distinct equivalent classes under Myhill-Nerode theorem meaning from the strings in each of these sets, we can append a string which will take the new string to L, while the same string appended to string in any other set would not reach L.

For example, for 000000, we append 11, for 0000000, we append 111 etc. So, in short we need to maintain the count of $1's$ and $0's$ and the count here is not finite.

D. This is regular. We need a finite automata with $5 \times 7 = 35$ states for maintaining the counts of $0's \mod 7$ and $1's \mod 5$ and there cannot be more than 35 possibilities for this. With each input symbol, the transition must be going to one among these.

Correct Answer: C

56 votes -- Arjun Suresh (330k points)

5.17.13 Regular Languages: GATE CSE 2007 | Question: 31 top https://gateoverflow.in/1229

✔ Correct Option: **C**

A. CFL
B. CFL
C. Regular, language is string starting and ending with the same symbol and having length at least 3. e.g. $0x0$ or $1x1$
D. CFL

http://gatecse.in/wiki/Identify_the_class_of_the_language

References

34 votes -- Vikrant Singh (11.2k points)

5.17.14 Regular Languages: GATE CSE 2007 | Question: 7 top https://gateoverflow.in/1205

✔ Correct Option: **B**

Every finite subset of a non-regular set is regular.

Any finite set is always regular.

Σ^* being regular any non regular language is a subset of this, and hence (A) is false.

If we take a CFL which is not regular, and takes union with its complement (complement of a CFL which is not regular won't be regular as regular is closed under complement), we get Σ^* which is regular. So, (C) is false.

Regular set is not closed under infinite union. Example:
Let $L_i = \{0^i 1^i\}, i \in N$

Now, if we take infinite union over all i, we get

$L = \{0^i 1^i \mid i \in N\}$, which is not regular.

So, (D) is false.

45 votes -- Omesh Pandita (1.9k points)

5.17.15 Regular Languages: GATE CSE 2008 | Question: 53 top https://gateoverflow.in/476

✔ Answer is **A**.

Since in option 2 and 3, n is dependent on m, therefore a comparison has to be done to evaluate those and hence are not regular.

I and IV are clearly regular sets.

24 votes -- Gate Keeda (15.9k points)

5.17.16 Regular Languages: GATE CSE 2011 | Question: 24 top https://gateoverflow.in/3429

✔ Correct Option: **C**

complement of regular Language is regular

41 votes -- VOOTLA SRINIVAS (279 points)

5.17.17 Regular Languages: GATE CSE 2012 | Question: 25 top https://gateoverflow.in/1609

✔ $L = \{ab, aa, baa\}$

1. $abaabaaabaa = ab \;\; aa \;\; baa \;\; ab \;\; aa$ belong to L^* (combinations of strings in L)
2. $aaaabaaaa = aa \;\; aa \;\; baa \;\; aa$ belong to L^*
3. $baaaaabaaaab = baa \;\; aa \;\; ab \;\; aa \;\; aa$ **b** does not belong to L^*
4. $baaaabaa = baa \;\; aa \;\; ab \;\; aa$ belong to L^*

Correct Answer: C

31 votes -- Praveen Saini (41.9k points)

5.17.18 Regular Languages: GATE CSE 2013 | Question: 8 top https://gateoverflow.in/1417

✔ *Concatenation of empty language with any language will give the empty language and* $L_1{}^* = \phi^* = \epsilon$.

Therefore,

$L_1L_2^* \cup L_1^*$
$= \phi.\,(L_2)^* \cup \phi^*$
$= \phi \cup \{\epsilon\}\; (\because \phi$ concatenated with anything is ϕ and $\phi^* = \{\epsilon\})$
$= \{\epsilon\}$.

Hence, option (A) is true.

PS: $\phi^* = \epsilon$, *where* ϵ *is the regular expression and the language it generates is* $\{\epsilon\}$.

72 votes -- Praveen Saini (41.9k points)

5.17.19 Regular Languages: GATE CSE 2014 Set 1 | Question: 15 top https://gateoverflow.in/1781

✔ **(A)** is **CFL** and **(B)** and **(D)** are **CSL**.

(C) is regular and regular expression for **(C)** would be:

$$a^*b(a^*ba^*ba^*b)^+a^*$$

Correct Answer: C

28 votes -- Arjun Suresh (330k points)

5.17.20 Regular Languages: GATE CSE 2014 Set 2 | Question: 15 top https://gateoverflow.in/1971

✔ **Option A.**

$L_1 = \{\varepsilon, a, aa, aaa, aaaa, \ldots\}$

$L_2 = \{\varepsilon, b, bb, bbb, bbbb, \ldots\}$

$$L_1 \cdot L_2 = \left\{ \begin{array}{cccccc} \varepsilon, & & & & & \\ a, & b, & & & & \\ aa, & ab, & bb & & & \\ aaa, & aab, & abb, & bbb, & & \\ aaaa, & aaab, & aabb, & abbb, & bbbb, & \ldots \end{array} \right\}$$

$L_1 \cdot L_2 = a^*b^*$

Thus, $L_1 \cdot L_2$ is Regular.

(Also, since both L_1 and L_2 are Regular, their concatenation has to be Regular since Regular languages are closed under concatenation)

However, $L_1 \cdot L_2 \neq a^n b^n$. This is because in $a^* b^*$, the number of a's and b's can be different whereas in $a^n b^n$ they have to be the same.

58 votes -- Viral Kapoor (1.9k points)

5.17.21 Regular Languages: GATE CSE 2014 Set 2 | Question: 36 top https://gateoverflow.in/1995

✔ Correct Option: **A**

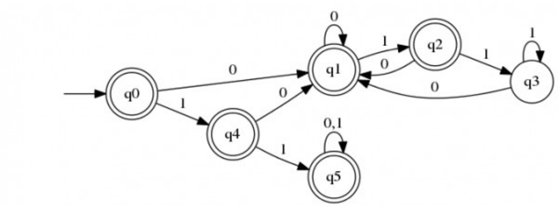
Though at first look both L_1 and L_2 looks non-regular, L_1 is in fact regular. The reason is the relation between 110 and 011.

We cannot have two $110'$s in a string without a 011 or vice verse. And this would mean that we only need a finite number of states to check for acceptance of any word in this language.

That was just an intuitive explanation. Now I say that L contains all binary strings starting with 11. Yes, if a binary string starts with 11, it can never have more no. of 011 than 110.

Lets take an example:
$11\ 011\ 011$ -There are two $011'$s. But there are also two 110's. Similarly for any binary string starting with 11.

Using this property, DFA for L_1 can be constructed as follows:



122 votes -- gatecse (62.6k points)

5.17.22 Regular Languages: GATE CSE 2015 Set 2 | Question: 51 top https://gateoverflow.in/8254

✔ Answer is **A**.

L_1: all strings of length 3 or more with same start and end letter- because everything in middle can be consumed by x as per the definition of L.

L_2: We need to compare number of a's and b's and these are not bounded. So, we need at least a **DPDA.**

L_3: Any number of a's followed by any number of b's followed by any number of c's. Hence regular.

41 votes -- Vikrant Singh (11.2k points)

5.17.23 Regular Languages: GATE CSE 2016 Set 2 | Question: 17 top https://gateoverflow.in/39542

✔ Answer is **C.**

$S_1 \to aS_1b \mid \epsilon$

$L_1 = \{a^n b^n \mid n \geq 0\}$ is CFL

$S_2 \to abS_2 \mid \epsilon$

$L_2 = \{(ab)^n \mid n \geq 0\}$ is Regular having regular expression $(ab)^*$

51 votes -- Praveen Saini (41.9k points)

5.17.24 Regular Languages: GATE CSE 2018 | Question: 52 top https://gateoverflow.in/204127

✔ $L_1 = \{\epsilon + 0(00)^*\}$

Now that is given language L, we have find order of it.

$L^0 = \{\epsilon\}$

$L^1 = L^0.L = \{\epsilon\}.\{\epsilon + 0(00)^*\} = \{\epsilon + 0(00)^*\}$

$L^2 = L^1.L = \{\epsilon + 0(00)^*\}.\{\epsilon + 0(00)^*\}$

$= \{\epsilon + 0(00)^* + 0(00)^*0(00)^*\}$

$L^2 = \{0^*\}$

$L^3 = L^2.L = \{0^*\}.\{\epsilon + 0(00)^*\}$

$= \{0^* + 0^*0(00)^*\}$

$L^3 = \{0^*\}$

Order of L is 2 such that $L^2 = L^{2+1}$

81 votes -- Praveen Saini (41.9k points)

5.17.25 Regular Languages: GATE CSE 2019 | Question: 7 top https://gateoverflow.in/302841

✔ ww^R is well known CFL - the PDA can non-deterministically determine the middle position of the string and start popping (this is not DCFL though).

Reverse, Suffix, Prefix, Concatenation of Regular(s) is Regular. **Answer is (B)**.

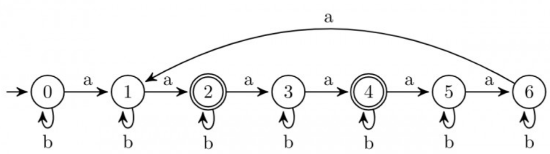
27 votes -- Digvijay (44.9k points)

5.17.26 Regular Languages: GATE CSE 2020 | Question: 51 top https://gateoverflow.in/333180

✔ Number of $a's$ in x are divisible by 2 but not divisible by 3 means number of $a's$ can be $2, 4, 8, 10, 14, 16, 20, 22, 26, \dots$

Terms which are divisible by both 2 and 3 like $6, 12, 18, 24, \dots$ are dropped from the above sequence at finite interval in $*$ position from above sequence $2, 4, *, 8, 10, *, 14, 16, *, 20, 22, *, 26, 28, *, 32, 34, \dots$

Here, sequence $\{6, 12, 18, 24, \dots\}$ is in AP. Now, to drop this sequence from our original sequence which represent the number of $a's$, we have to go through at least total 7 states which says total 6 $a's$ have seen so far and since terms of this sequence come at regular interval, So, we will make a cycle in the DFA as :



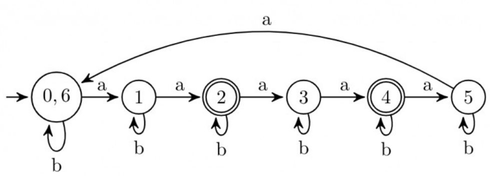
Now, observe that states 0 and 6 are equivalent because by definition, 2 states p and q are equivalent i.e.

$$p \approx q \overset{def}{\Leftrightarrow} \forall x \in \Sigma^* (\hat{\delta}(p, x) \in F \Leftrightarrow \hat{\delta}(q, x) \in F)$$

It says 2 states are equivalent if for all strings, they both either go to final state or both go to non-final state means they have the same nature.

So, here, we can merge states 0 and 6. Other states can't be merged because they are in cycle and if we merge states which are in cycle, machine will not accept the desired number of $a's$ and reject $6, 12, 18, \dots a's$.

So, our final collapsing DFA is :



9 votes -- ankitgupta.1729 (15k points)

5.17.27 Regular Languages: GATE CSE 2020 | Question: 8 top

☛ https://gateoverflow.in/333223

✔ Keeping L_2 as Σ^*, what ever may be L_1,, we get a Regular language.

So, statement I is wrong.

If regular languages are closed under infinite union, then $L = \{a^n.b^n | n > 0\}$ must be regular as it is equal to $\{ab\} \cup \{aabb\} \cup \{aaabbb\} \cup \dots$

So, statement II is wrong.

Option D is correct.

11 votes

-- Shaik Masthan (50.4k points)

5.17.28 Regular Languages: GATE CSE 2021 Set 2 | Question: 9 top

☛ https://gateoverflow.in/357531

✔ Answer : Option C.

The question is asking about Number of states in minimal DFA.

If L is regular then so is $L - \{01\}$, so is $L \cup \{01\}$, so is $L.L$, so is $\{0,1\}^* - L$.

But if minimal DFA for L has k states then can we guarantee that minimal DFA for $L - \{01\}$ will have k states ?? can we guarantee that minimal DFA for $L \cup \{01\}$ will have k states ?? can we guarantee that minimal DFA for $L.L$ will have k states ?? can we guarantee that minimal DFA for complement of L will have k states ??

First we will check Option C. We will prove that if minimal DFA for a regular language L has n states then the minimal DFA for complement of L will also have n states.

Since L is regular language, so, we have some DFA D that accepts L.

We can describe D as following : $D(Q, \Sigma, \delta, q_0, F)$

In this DFA D, If we make the accepting states be non-accepting, and make the non-accepting states be accepting, then this new automata D' can be described as $D'(Q, \Sigma, \delta, q_0, Q - F)$ (Because in D', set of final states is $Q - F$) and this D' has following properties :

1. D' is a DFA (Because we are not changing the transition function so for every state, on every alphabet symbol we still have exactly one transitions)

2. Since D, D' have same states, same initial state, same transition function, So, on any string w, both D, D' will go to same state, say, q. Now, we have two cases :

- if q is final state in D then q is non-final in D', So, $w \in L(D)$ and $w \notin L(D')$
- if q is non-final state in D then q is final in D', So, $w \notin L(D)$ and $w \in L(D')$

So, any string w, either it belongs to $L(D)$ or to $L(D')$ But Not to both. So, $L(D)$ and $L(D')$ are complement of each other.

So, the conclusion is that :

If a DFA D accepts language L, then DFA D' will accept language L', where D' is constructed from D by changing the final states to Non-final and vice versa.

So far we have proven that If D is DFA for L then D' is DFA for L'.

Now, the second part is to prove that :

If D is minimal DFA for L then D' is minimal DFA for L'.

This is easy to prove by contradiction.

Let DFA D be the minimal DFA of L with n states in it.

For contradiction, let us assume that DFA D' is Not minimal DFA for L' then it means that L' has some minimal DFA M in which we have less than n states.

Now, we construct M' by swapping final and non-final states in M, So, M' will accept complement of L' i.e. M' accepts L. So, now we have a DFA (M') for L in which we have less than n states. But this is contradiction because minimal DFA for L has n states.

So, our assumption is false i.e. It is Not the case that DFA D' is Not minimal DFA for L'.

So, D' is minimal DFA for L'.

So, we have proven that :

If a D is a minimal DFA for a regular language L then D' is minimal DFA for L'.
Since D and D' have same number of states, so we can say that number of states in minimal DFA for regular language L and number of states in minimal DFA for complement of L is same.

Option A is false :
For counter example, take $L = \{01\}$, minimal DFA for L has 4 states. But minimal DFA for $L - \{01\}$ has 1 state only.
Option B is false :
For counter example, take $L = \{\}$, minimal DFA for L has 1 state. But minimal DFA for $L \cup \{01\}$ has 4 states.
Option D is false :
For counter example, take $L = \{0\}$, minimal DFA for L has 3 states. But minimal DFA for $L.L$ has 4 states.

6 votes -- Deepak Poonia (23.3k points)

5.17.29 Regular Languages: GATE IT 2006 | Question: 30 top https://gateoverflow.in/3569

✔ Option **C** is not True.

A. **Every language has a regular superset:** True. Σ^* is such a superset.
B. **Every language has a regular subset:** True. $\emptyset$ is such a subset.
C. **Every subset of a regular language is regular:** False. $a^n b^n \subset \Sigma^*$, but $a^n b^n$ is not Regular.
D. **Every subset of a finite language is regular:** True. Every subset of a finite set must be finite by definition. Every finite set is regular. Hence, every subset of a finite language is regular.

54 votes -- Pragy Agarwal (18.3k points)

5.17.30 Regular Languages: GATE IT 2006 | Question: 80 top https://gateoverflow.in/3624

✔ Correct answer is **B**. Only I.

Repeat $(L) = \{ww \mid w \in L\}$ is non regular language

Half(L), suffix(L), and prefix(L) are regular languages

Reference:

https://gateoverflow.in/3637/gate2006-it_81

References

16 votes -- Praveen Saini (41.9k points)

5.17.31 Regular Languages: GATE IT 2006 | Question: 81 top https://gateoverflow.in/3637

✔

I. repeat(L) is non regular
http://www.cs.odu.edu/~toida/nerzic/390teched/regular/reg-lang/non-regularity.html [example 2]
II. prefix(L) is regular
http://www.public.asu.edu/~ccolbou/src/355hw2sols09.pdf [2(a)]
III. suffix(L) is regular
http://www.public.asu.edu/~ccolbou/src/355hw2sols09.pdf [2(b)]
IV. half(L) is regular
https://math.stackexchange.com/questions/1539658/automata-prove-that-if-l-is-regular-than-halfl-is-regular-too

So, in **first part of question**, option **B** is correct only (i).

For **second part of question:**

A is answer.

- For option B $L = \{\epsilon, a, ab, bab\}$, $\text{repeat}(L) = \{\epsilon, aa, abab, babbab\}$ is regular. So, not a suitable example.
- For option C, regular expression for Repeat(L) will be $(abab)^*$ and hence regular. So, this also is not a suitable example.
- For option D, L it self is not regular and hence it can not be the answer.
- For A, all strings of $\text{repeat}(\text{L})$ is in L. But that does not mean $\text{repeat}(L)$ is regular. For that we also have to ensure all strings in L is in $\text{repeat}(\text{L})$ which is not the case (or all strings not in $\text{repeat}(\text{L})$ is not in L). $\text{repeat}(\text{L})$ here is $\{ww \mid w \in (a+b)^*\}$ which is actually not even context-free.

References

42 votes -- Praveen Saini (41.9k points)

5.17.32 Regular Languages: GATE IT 2008 | Question: 35 top

https://gateoverflow.in/3345

✔ L_1 is regular.. since 10000 is finite. So, finite states are required.

L_3 is also regular. We can make DFA for L_3. States will represent mod 2 for 0 and mod 2 for 1, which is finite

L_2 is non regular..It is CFG $S \to aSa \mid \ldots \mid zSz \mid \epsilon \mid [a-z]$

So, option is(**D**).

29 votes -- Vicky Bajoria (4.1k points)

5.18 Turing Machine (4) top

5.18.1 Turing Machine: GATE CSE 2003 | Question: 53 top

https://gateoverflow.in/941

A single tape Turing Machine M has two states $q0$ and $q1$, of which $q0$ is the starting state. The tape alphabet of M is $\{0, 1, B\}$ and its input alphabet is $\{0, 1\}$. The symbol B is the blank symbol used to indicate end of an input string. The transition function of M is described in the following table.

	0	1	B
$q0$	$q1, 1, R$	$q1, 1, R$	$Halt$
$q1$	$q1, 1, R$	$q0, 1, L$	$q0, B, L$

The table is interpreted as illustrated below.

The entry $(q1, 1, R)$ in row $q0$ and column 1 signifies that if M is in state $q0$ and reads 1 on the current page square, then it writes 1 on the same tape square, moves its tape head one position to the right and transitions to state $q1$.

Which of the following statements is true about M?

A. M does not halt on any string in $(0+1)^+$
B. M does not halt on any string in $(00+1)^*$
C. M halts on all strings ending in a 0
D. M halts on all strings ending in a 1

gate2003-cse theory-of-computation turing-machine normal

Answer

5.18.2 Turing Machine: GATE CSE 2003 | Question: 54 top

https://gateoverflow.in/355

Define languages L_0 and L_1 as follows :

$L_0 = \{\langle M, w, 0 \rangle \mid M \text{ halts on } w\}$

$L_1 = \{\langle M, w, 1 \rangle \mid M \text{ does not halts on } w\}$

Here $\langle M, w, i \rangle$ is a triplet, whose first component M is an encoding of a Turing Machine, second component w is a string, and third component i is a bit.

Let $L = L_0 \cup L_1$. Which of the following is true?

A. L is recursively enumerable, but L' is not
B. L' is recursively enumerable, but L is not
C. Both L and L' are recursive
D. Neither L nor L' is recursively enumerable

theory-of-computation turing-machine gate2003-cse difficult

Answer

5.18.3 Turing Machine: GATE CSE 2004 | Question: 89 top

https://gateoverflow.in/1083

L_1 is a recursively enumerable language over Σ. An algorithm A effectively enumerates its words as $\omega_1, \omega_2, \omega_3, \dots.$ Define another language L_2 over $\Sigma \cup \{\#\}$ as $\{w_i \# w_j \mid w_i, w_j \in L_1, i < j\}$. Here # is new symbol. Consider the following assertions.

- S_1 : L_1 is recursive implies L_2 is recursive
- S_2 : L_2 is recursive implies L_1 is recursive

Which of the following statements is true?

A. Both S_1 and S_2 are true
B. S_1 is true but S_2 is not necessarily true
C. S_2 is true but S_1 is not necessarily true
D. Neither is necessarily true

gate2004-cse theory-of-computation turing-machine difficult

Answer

5.18.4 Turing Machine: GATE CSE 2014 Set 2 | Question: 35 top

https://gateoverflow.in/1994

Let $\langle M \rangle$ be the encoding of a Turing machine as a string over $\Sigma = \{0, 1\}$. Let

$$L = \{\langle M \rangle \mid M \text{ is a Turing machine that accepts a string of length } 2014\}.$$

Then L is:

A. decidable and recursively enumerable
B. undecidable but recursively enumerable
C. undecidable and not recursively enumerable
D. decidable but not recursively enumerable

gate2014-cse-set2 theory-of-computation turing-machine normal

Answer

Answers: Turing Machine

5.18.1 Turing Machine: GATE CSE 2003 | Question: 53 top

https://gateoverflow.in/941

✔ Correct Option: **A**

Or epsilon is only accepted i.e tape contain **B** as the first character.

23 votes -- Supromit Roy (633 points)

5.18.2 Turing Machine: GATE CSE 2003 | Question: 54 top

https://gateoverflow.in/355

✔ Both L and L’ are undecidable and not even semi-decidable (not recursively-enumerable). Because halting problem can be solved with both L and L’ .

Halting problem can be stated as follows: A machine M and a word w are given. You have to tell, if M halts on w.

So, to solve halting problem $\langle M, w \rangle$ using L, just give $\langle M, w, 0 \rangle$ and $\langle M, w, 1 \rangle$ to two instances of T which is the assumed

Turing machine for L. If T accepts the triplet $\langle M, w, 0\rangle$, it means M halts on w => we have solved halting problem. If T accepts the triplet $\langle M, w, 1\rangle$, it means M doesn't halt on w => we have solved halting problem. We know that either $\langle M, w, 0\rangle$ or $\langle M, w, 1\rangle$ is in L. So, if L is recursively enumerable, T is bound to stop on at least one of these inputs (TM for a recursively enumerable language stops and accepts, when provided with a word in its language).

Hence, if L is recursively enumerable we can solve halting problem $\Rightarrow L$ is not recursively enumerable.
Similarly, we can also show that halting problem can be solved with L'. (shown at end)

Hence, neither L nor L' is recursively enumerable.

To solve halting problem $\langle M, w\rangle$ using L', just give $\langle M, w, 0\rangle$ and $\langle M, w, 1\rangle$ to two instances of T' which is the assumed Turing machine for L'. If T' accepts the triplet $\langle M, w, 0\rangle$, it means M does not halt on w => we have solved halting problem. If T accepts the triplet $\langle M, w, 1\rangle$, it means M halt on w => we have solved halting problem. We know that either $\langle M, w, 0\rangle$ or $\langle M, w, 1\rangle$ is in L'. So, if L' is recursively enumerable, T' is bound to stop on at least one of these inputs (TM for a recursively enumerable language stops and accepts, when provided with a word in its language).
Hence, if L' is recursively enumerable we can solve halting problem $\Rightarrow L'$ is not recursively enumerable.

PS: If the bit part of the triplet is absent then L_0 is halting problem and L_1 is its complement and $L_0 \cup L_1 = \Sigma^*$, which is regular. Lets see how it happens.

Let the alphabet set be $\{0, 1\}$. Now for any string like 0010101 there are only two options- belong to L or belong to L' as this is what complement says. Now, lets take the case for L_0 and a string $001 \dots 10 - 01 - 1$, ("-" shown for notation purpose only) where the first component describes a TM M followed by input "$w = 01$" and last bit "1". Now suppose M halts on "01". Still the given input is not in L_0 as the last bit is "1" and not "0" as required by L_0. So, this input must be in L_0'. But since M halts on w, this input is not in L_1 either. Similarly, we can get an infinite set of strings which does not belong to both L_0 and L_1 and this makes their union not Σ^* but an irregular (not r.e. as proved earlier) set. If the last bit is removed from the definition of L_0 and L_1, then any string should be present in either L_0 or L_1 and their union would be Σ^*.

Correct Answer: D

77 votes -- gatecse (62.6k points)

5.18.3 Turing Machine: GATE CSE 2004 | Question: 89 top https://gateoverflow.in/1083

✔ S_1 is TRUE.

If L_1 is recursive L_2 must also be recursive. Because to check if a word $w = w_i \# w_j$ belong to L_2, we can give w_i and w_j to the decider for L_1 and if both are accepted then w belong to L_1 and not otherwise.

S_2 is TRUE.

With a decider for L_2 we can make a decider for L_1 as follows. Let w_1 be the first string enumerated by algorithm A for L_1. Now, to check if a word w belongs to L_1, make a string $w' = w_1 \# w$ and give it to the decider for L_2 and if accepted, then w belongs to L_1 and not otherwise.

So, answer must be **A**.

PS: For the second part, the given construction can fail if L_1 happens to be a finite language (more specifically if L_1 is empty). But all finite languages are anyway decidable.

43 votes -- Arjun Suresh (330k points)

5.18.4 Turing Machine: GATE CSE 2014 Set 2 | Question: 35 top https://gateoverflow.in/1994

✔ There are only a finite number of strings of length 2014. So, we can give all those strings to TM simulating each string for 1 step, then 2 step and so on (dovetailing), and if the TM accepts any of them ("yes" case of TM), we can say "yes". So, L is recursively enumerable.

(If the TM doesn't accept any string of length 2014, it can go to an infinite loop ("no" case of TM), and hence we can't say the method is decidable).

Now, to prove whether the problem is decidable or not we can make use of Rice's theorem. Rice's theorem (I) states that any non-trivial property of $L(TM)$ is undecidable. $L(TM)$ has a string of length 2014 is a non-trivial property as there are TMs whose language contains such a string and there are TMs whose language doesn't have such a string. So, the given

problem is undecidable.
http://gatecse.in/wiki/Rice%27s_Theorem_with_Examples

Correct Answer: B

References

👍 104 votes -- Arjun Suresh (330k points)

Answer Keys

5.1.1	A;C	5.1.2	N/A	5.1.3	A	5.1.4	C	5.1.5	D
5.1.6	B	5.1.7	B	5.1.8	D	5.2.1	B	5.2.2	False
5.2.3	A;D	5.2.4	C	5.2.5	A	5.2.6	B	5.2.7	B
5.2.8	B	5.2.9	N/A	5.2.10	N/A	5.2.11	B	5.2.12	C
5.2.13	B	5.2.14	D	5.2.15	B	5.2.16	B	5.2.17	D
5.2.18	D	5.2.19	B	5.2.20	B	5.2.21	D	5.2.22	A
5.2.23	C	5.2.24	C	5.2.25	C	5.2.26	B;C;D	5.2.27	B
5.2.28	B	5.2.29	A	5.2.30	C	5.2.31	D	5.2.32	B
5.2.33	D	5.3.1	B	5.3.2	C	5.3.3	B	5.4.1	True
5.4.2	True	5.4.3	N/A	5.4.4	B;D	5.4.5	B;C	5.4.6	N/A
5.4.7	D	5.4.8	B	5.4.9	N/A	5.4.10	A	5.4.11	B
5.4.12	N/A	5.4.13	N/A	5.4.14	C	5.4.15	C	5.4.16	B
5.4.17	B	5.4.18	D	5.4.19	D	5.4.20	A	5.4.21	D
5.4.22	C	5.4.23	C	5.4.24	A	5.4.25	D	5.4.26	D
5.4.27	A	5.4.28	C	5.5.1	N/A	5.5.2	N/A	5.5.3	N/A
5.5.4	True	5.5.5	B	5.5.6	N/A	5.5.7	N/A	5.5.8	A
5.5.9	N/A	5.5.10	A	5.5.11	N/A	5.5.12	C	5.5.13	B
5.5.14	A	5.5.15	B	5.5.16	B	5.5.17	C	5.5.18	A
5.5.19	C	5.5.20	A	5.5.21	C	5.5.22	B	5.5.23	D
5.5.24	D	5.5.25	A	5.5.26	B	5.5.27	C	5.5.28	D
5.5.29	256 : 256	5.5.30	B	5.5.31	B	5.5.32	A	5.5.33	B
5.5.34	D	5.5.35	A	5.5.36	D	5.5.37	A	5.5.38	A
5.5.39	A	5.5.40	A	5.5.41	A	5.6.1	C	5.7.1	C
5.7.2	N/A	5.7.3	N/A	5.7.4	N/A	5.7.5	A;C	5.7.6	B
5.7.7	B	5.7.8	B	5.7.9	A	5.7.10	D	5.7.11	B
5.7.12	B	5.7.13	D	5.7.14	C	5.7.15	D	5.7.16	C
5.7.17	D	5.7.18	C	5.7.19	B	5.7.20	D	5.7.21	B
5.7.22	C	5.7.23	A	5.7.24	B;C;D	5.7.25	D	5.7.26	A
5.7.27	B	5.8.1	0	5.8.2	B	5.8.3	N/A	5.8.4	N/A
5.8.5	B	5.8.6	N/A	5.8.7	B	5.8.8	B	5.8.9	D

5.8.10	B	5.8.11	D	5.8.12	A	5.8.13	B	5.8.14	C
5.8.15	B	5.8.16	A	5.8.17	1	5.8.18	3	5.8.19	B
5.8.20	2	5.8.21	4	5.8.22	8	5.8.23	D	5.8.24	120
5.8.25	A	5.9.1	A;C	5.9.2	C	5.9.3	B	5.9.4	D
5.9.5	D	5.9.6	B	5.9.7	C	5.10.1	D	5.10.2	A
5.10.3	C	5.10.4	B	5.10.5	A	5.11.1	D	5.11.2	D
5.12.1	N/A	5.12.2	A	5.12.3	N/A	5.12.4	A	5.12.5	N/A
5.12.6	N/A	5.12.7	B	5.12.8	D	5.12.9	50 : 50	5.12.10	B
5.12.11	D	5.12.12	B	5.12.13	C	5.13.1	A;D	5.13.2	A
5.13.3	D	5.13.4	B	5.13.5	D	5.13.6	D	5.13.7	B
5.13.8	C	5.13.9	D	5.13.10	D	5.13.11	C	5.13.12	C
5.13.13	D	5.14.1	A	5.15.1	N/A	5.15.2	A;C	5.15.3	A
5.15.4	N/A	5.15.5	C	5.15.6	C	5.15.7	D	5.15.8	C;D
5.15.9	D	5.15.10	N/A	5.15.11	C	5.15.12	A	5.15.13	C
5.15.14	B	5.15.15	B	5.15.16	3	5.15.17	B	5.15.18	X
5.15.19	A;B;C	5.15.20	C	5.15.21	B	5.15.22	A	5.15.23	C
5.15.24	C	5.16.1	N/A	5.16.2	C	5.16.3	A	5.17.1	True
5.17.2	False	5.17.3	B;C	5.17.4	D	5.17.5	C	5.17.6	D
5.17.7	B	5.17.8	N/A	5.17.9	X	5.17.10	A	5.17.11	D
5.17.12	C	5.17.13	C	5.17.14	B	5.17.15	A	5.17.16	C
5.17.17	C	5.17.18	A	5.17.19	C	5.17.20	A	5.17.21	A
5.17.22	A	5.17.23	C	5.17.24	2	5.17.25	B	5.17.26	6
5.17.27	D	5.17.28	C	5.17.29	C	5.17.30	B	5.17.31	A
5.17.32	D	5.18.1	A	5.18.2	D	5.18.3	A	5.18.4	B

Since GATE Overflow (GO) started in August 2014, a lot of people have dedicated their time and effort in bringing this book now. Initiated by Omesh Pandita and Arjun Suresh as a Q/A platform for CSE students, Kathleen Bankson was instrumental in getting all previous year GATE questions here. Then experts like Praveen Saini, Happy Mittal, Sankaranarayanan P.N., Suraj Kumar etc. have contributed a lot to the answers in GATE Overflow. Pragy Agarwal even after topping GATE has continuously contributed to GATE Overflow with his knowledge as well as in making the contents beautiful with fine latex skills. Shaik Masthan and Deepak Poonia are other GATE toppers who have contributed to make the GATE Overflow contents error-free. We also have to thank the work done by our editors Jotheeswari Arasakumar, Lakshman Patel, Soujanya Reddy, Sabiha Banu, Misbah Ghaya, Ishrat Jahan, Nataliyah Ahmad, Kanza Bayad, Andrijana Milicevik and Pavan Singh who are continuously adding and keeping the contents in GATE Overflow neat and clean. There are also many toppers of GATE 2015-21 who are contributing a lot in GATE Overflow. The list of all the contributors can be found on GATE Overflow site but even that does not include the contributions of some like Arif Ali Anapparakkal for the book design, Arvind Devaraj and others for guidance and help, Aravind S.R. for fixing the page numbering issue, Marlyn Joseph for book printing support, Ajay K. Soni and Akshat Joshi for book design support and finally Shiva Sagar Rao, Hira Thakur and Ankit Singh for flagging and correcting the errors on GATE Overflow site which is reflected in this book also. Our final proof reading and editing were done with the help of Soujanya Reddy, Lakshman Patel, Ajay K. Soni, Lakshman Patel, M. Chandrakiruthiga, Pooja Khatri, Shikha Mallick, Manoja Rajalakshmi Aravindakshan, Subarna Das and Sukanya Das. Special mention to Ajay K. Soni, Lakshman Patel, Aditya Ravishankar, Sweta Kumari Rawani, Gyanendra Singh, Sabiha Banu, Sukanya Das, Subarna Das and Pooja Khatri for handling all the difficult images in Tikz.

From the GATE Overflow family, we thank the contributions of Silpa V.S., Digvijay Pandey, Rahul Kumar Yadav and others for getting the GATECSE Last-Rank page maintained. Bikram Ballav is behind most of the free exams on GATE Overflow and our exam interface which is already used for 50,000+ exam takes is contributed by Arindam Sarkar. Pragy Agarwal is also behind the rank and score predictor tool, used by GATE Overflow which has 99-100% accuracy over the last 6 years. Special thanks to Sachin Mittal for making the How to Use GATE Overflow videos, Silpa V.S. for classifying the questions topic-wise for the book, Pooja Palod for making our first GATE schedule and Debashish Deka, Priti Sharma and Digvijay Pandey for GATE Overflow classroom contributions. We also thank all the GATE toppers who took time to write a review for GATE Overflow. Finally on behalf of GATE Overflow family and all GATE aspirants we thank the wonderful faculties of IISc., IITs and other premier institutes who are behind the creation of such wonderful questions.

Without mentioning the great Q2A platform Question2Answer (on which GATE Overflow runs) our acknowledgement would not be complete. Q2A is an open source platform created by Gideon Greenspan and currently being maintained by Scott Vivian and numerous other contributors like NoahY whose open source plugin is used for this book creation. We express our sincere gratitude towards them.